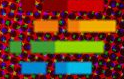
NEXT-GENERATION MICROSERVICES

PRINCIPLES, PRACTICES & REIMAGINED RULES OF ENGAGEMENT FOR UPSTARTS, MASTERS & EVERY FLAVOR OF SUPER HERO

BURR SUTTER
DIRECTOR, DEVELOPER EXPERIENCE
RED HAT
@BURRSUTTER

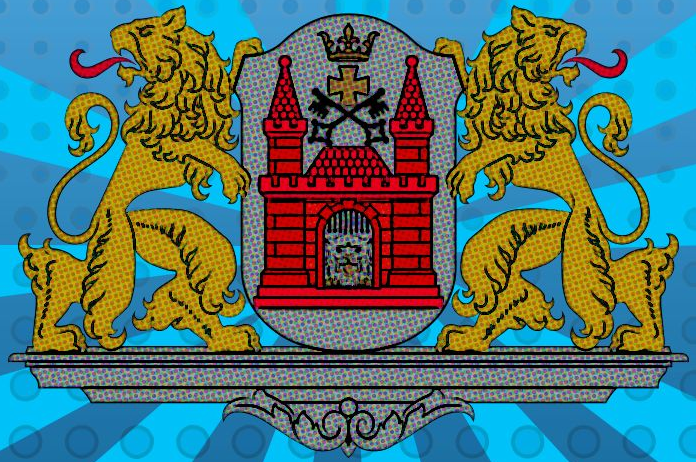
RIGA DEVDAYS 2019

bit.ly/RigaBurr2019



RED HAT DEVELOPER

MEANWHILE, IN THE LAND OF SUPER-FAST INTERNET ...



@burrsutter

RED HAT DEVELOPER

@burrsutter
RED HAT
DEVELOPER

THEY CAME BEARING GIFTS ... BOOKS ... MADE OF PAPER!

(OR BETTER YET, GO TO)

BIT.LY/ISTIOBOOK

@burrsutter

RED HAT DEVELOPER

MORE RESOURCES AVAILABLE FROM DEVOXX BELGIUM 2018

DEVOXX™ BELGIUM

HTTP://BIT.LY/2XIWXPG

"EXCELLENT STUFF.. WORTH 3 HOURS SPENDING TO WATCH THIS VIDEO.. REALLY LOVED IT. WILL GIVE A COMPLETE PICTURE ON KUBERNETES AND HOW IT WORKS." - PRADEEP

WHAT DO YOU SEE & HEAR IN YOUR OWN ORGANIZATION?

ARE CUSTOM SOFTWARE APPS/APIS A KEY STRATEGIC ADVANTAGE FOR YOUR ORGANIZATION?

DO YOU REGARD IT AS A COST CENTER THAT MUST BE BETTER STREAMLINED?

@burrsutter

RED HAT DEVELOPER

Journey to Awesomeness

DevOps

Self-Service, On-Demand, Elastic Infrastructure

Automation

CI & CD Deployment Pipeline

Advanced Deployment Techniques

Next-Generation Microservices

@burrsutter

RED HAT® DEVELOPER

Force Multipliers & Superpowers

Wonder Woman Style

Boomerang Tiara	Indestructible Bracelets	Lasso of Truth	Amazonian Shield	Hand-to-Hand Combat	Superhuman Strength
Re-Org to DevOps	Self-Service, On-Demand, Elastic Infrastructure	Automation Puppet, Chef, Ansible, Kubernetes	CI & CD Deployment Pipeline	Advanced Deployment Techniques	Microservices

@burrsutter

RED HAT® DEVELOPER

12 'Ways' of the Cloud-Native Warrior

1. DevOps & Agile
2. Skill Portability
3. On-Demand - Never Wait & Always Automate
4. Pipeline Throughput, Deployment Interval, Optimize for Learning
5. Deployment Techniques
6. Application Architecture vs Application Infrastructure
7. Linux Containers
8. Kubernetes/OpenShift & Hybrid Cloud
9. Service Mesh: Observability, Security, Traffic Control (Istio)
10. Serverless Serving (Knative)
11. Build In Cluster (Tekton)
12. Beyond HTTP (AMQP, Kafka, Knative Eventing, Camel-K)

@burrsutter

RED HAT DEVELOPER

Adapt what is useful, reject what is useless, and add what is specifically your own.

— *Bruce Lee*

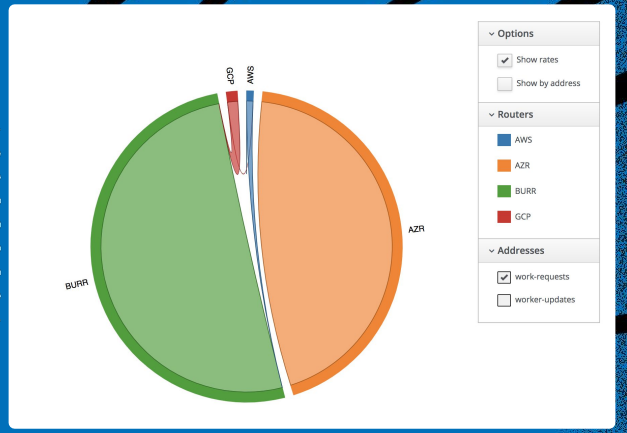
RED HAT® DEVELOPER

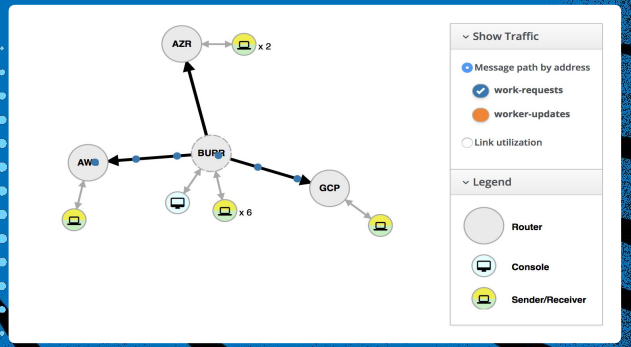
OpenShift
Message Processor (Worker)
AMQ

OpenShift
Message Processor (Worker)
AMQ

OpenShift
Message Processor (Worker)
AMQ

OpenShift
Message Processor (Worker)
AMQ

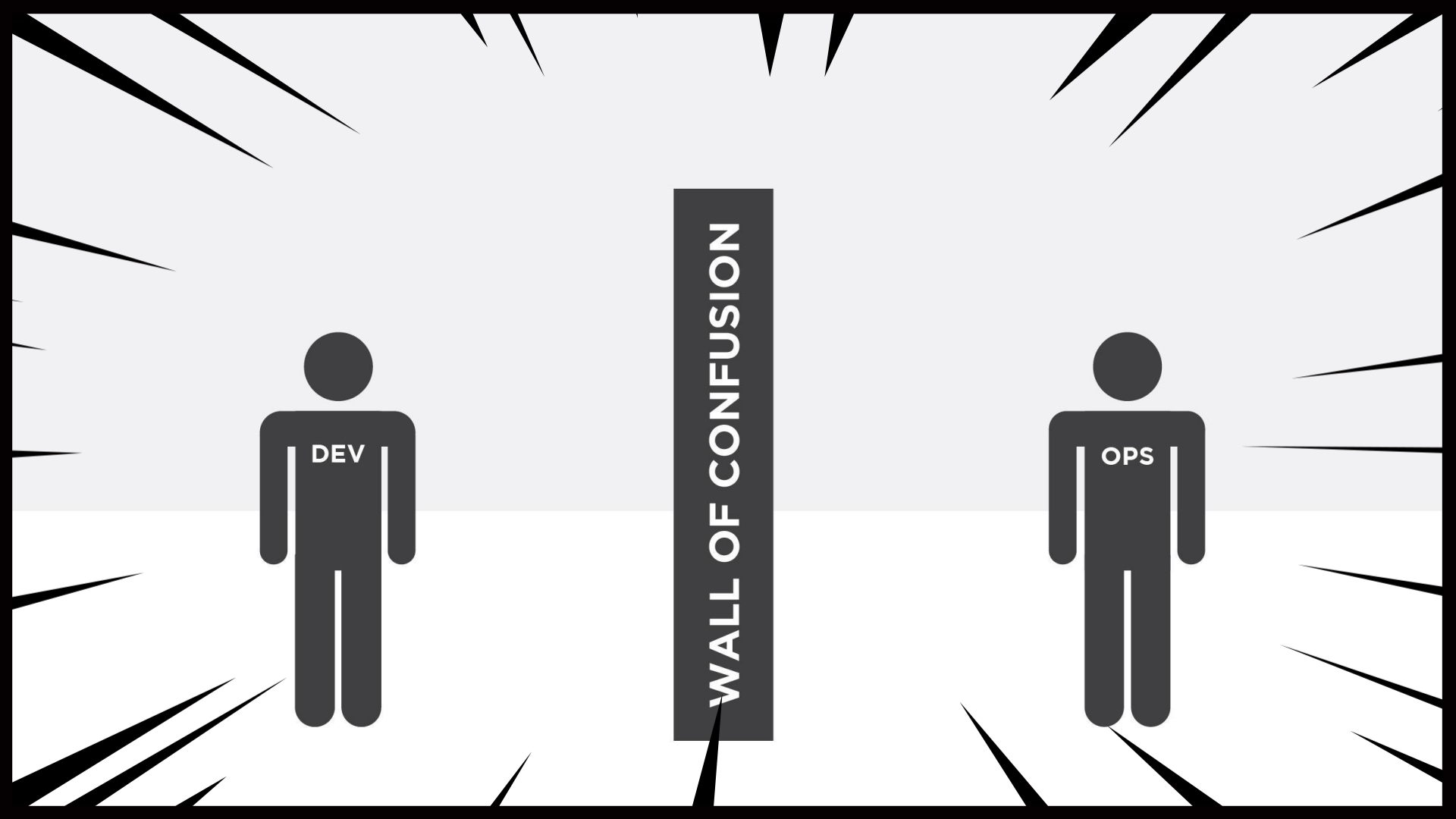


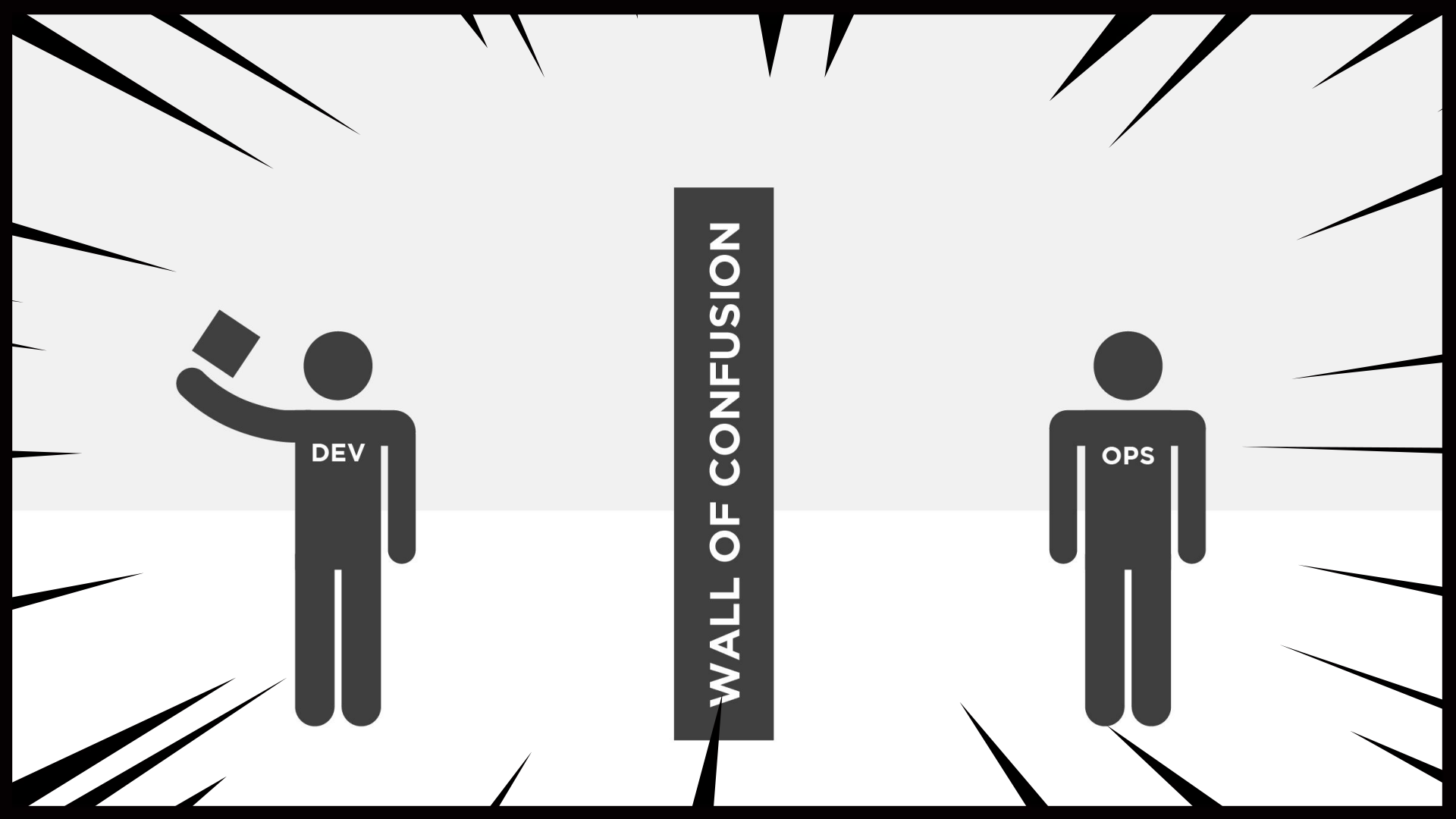


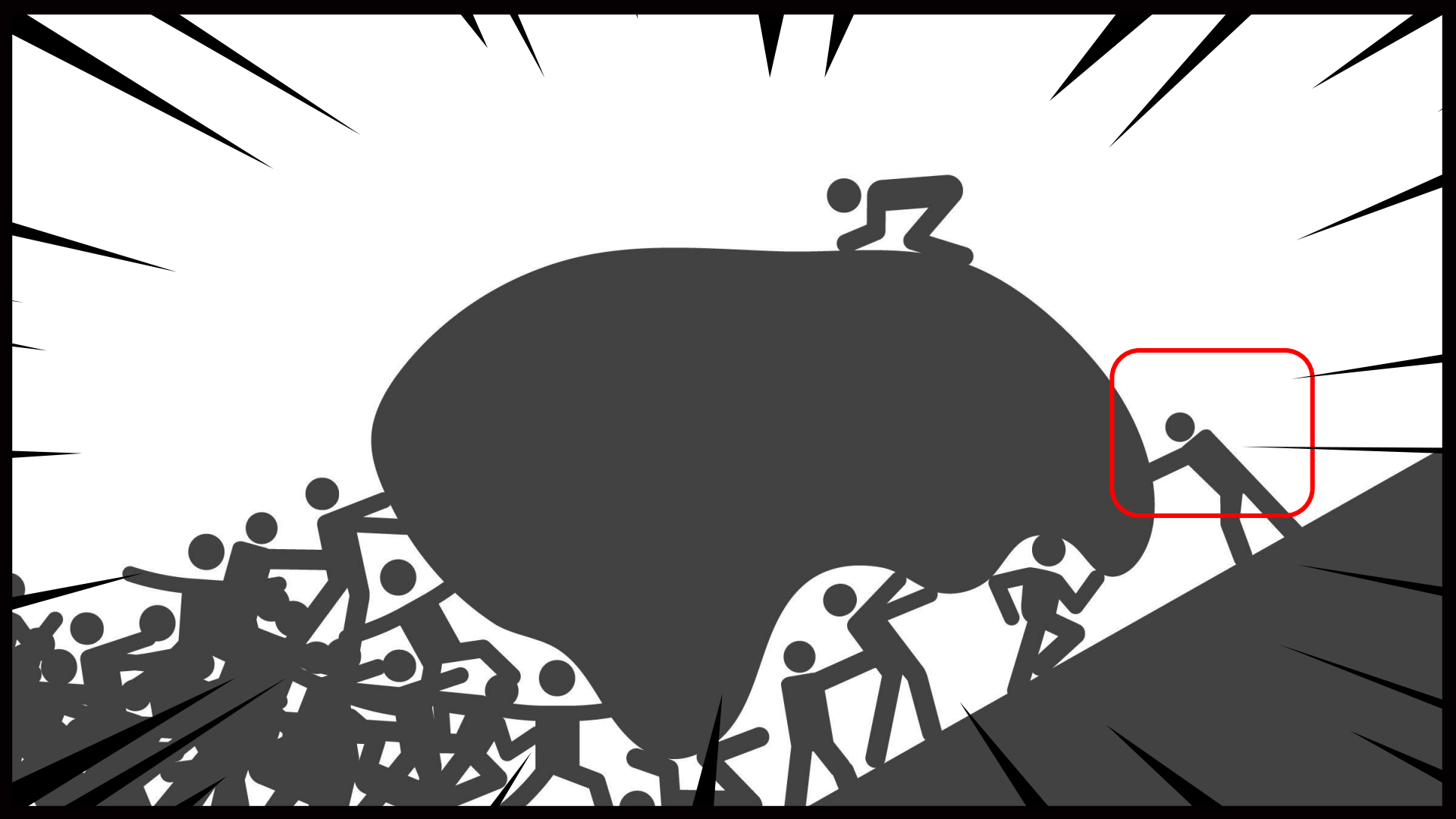
@burrsutter

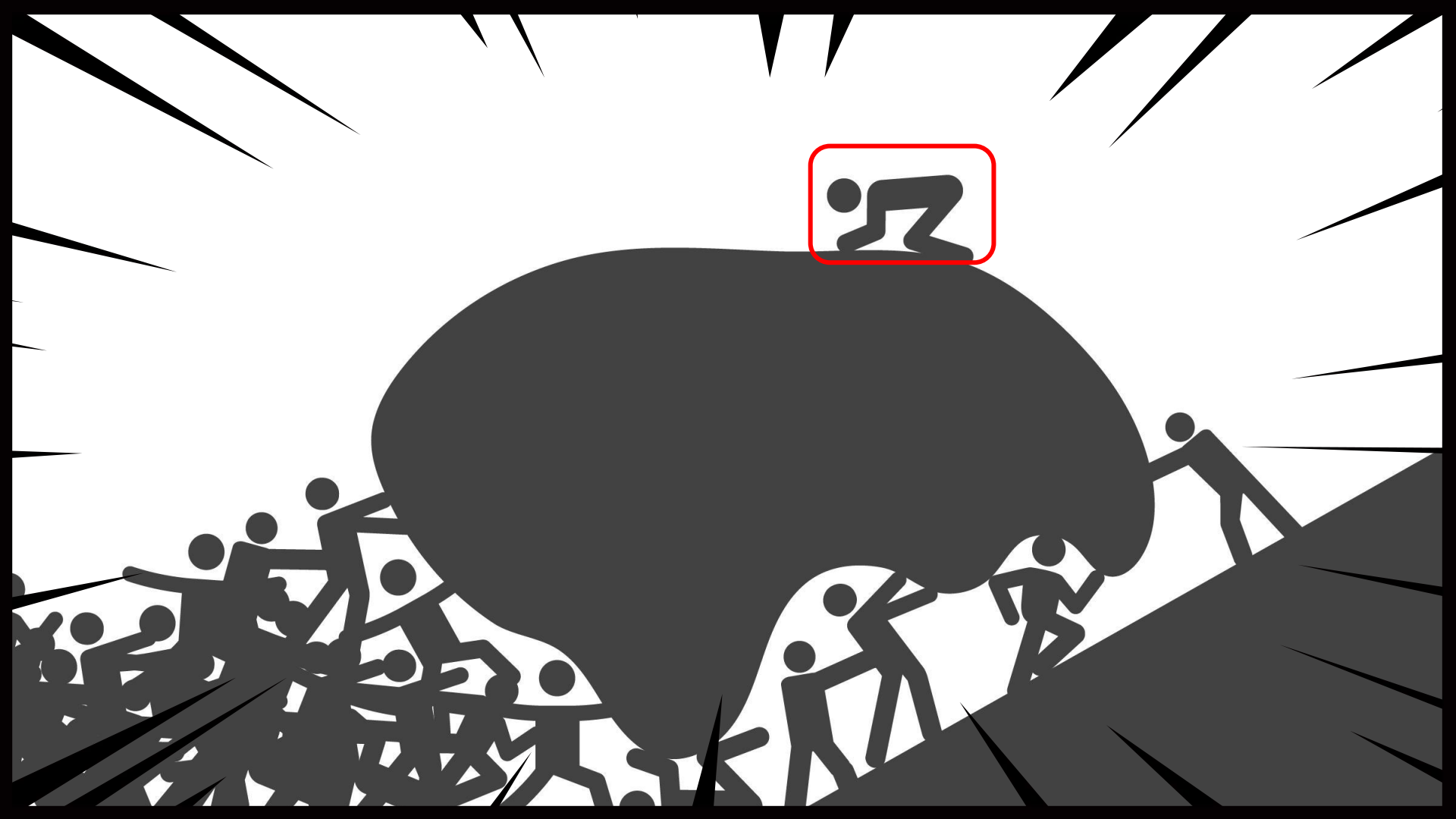
RED HAT DEVELOPER

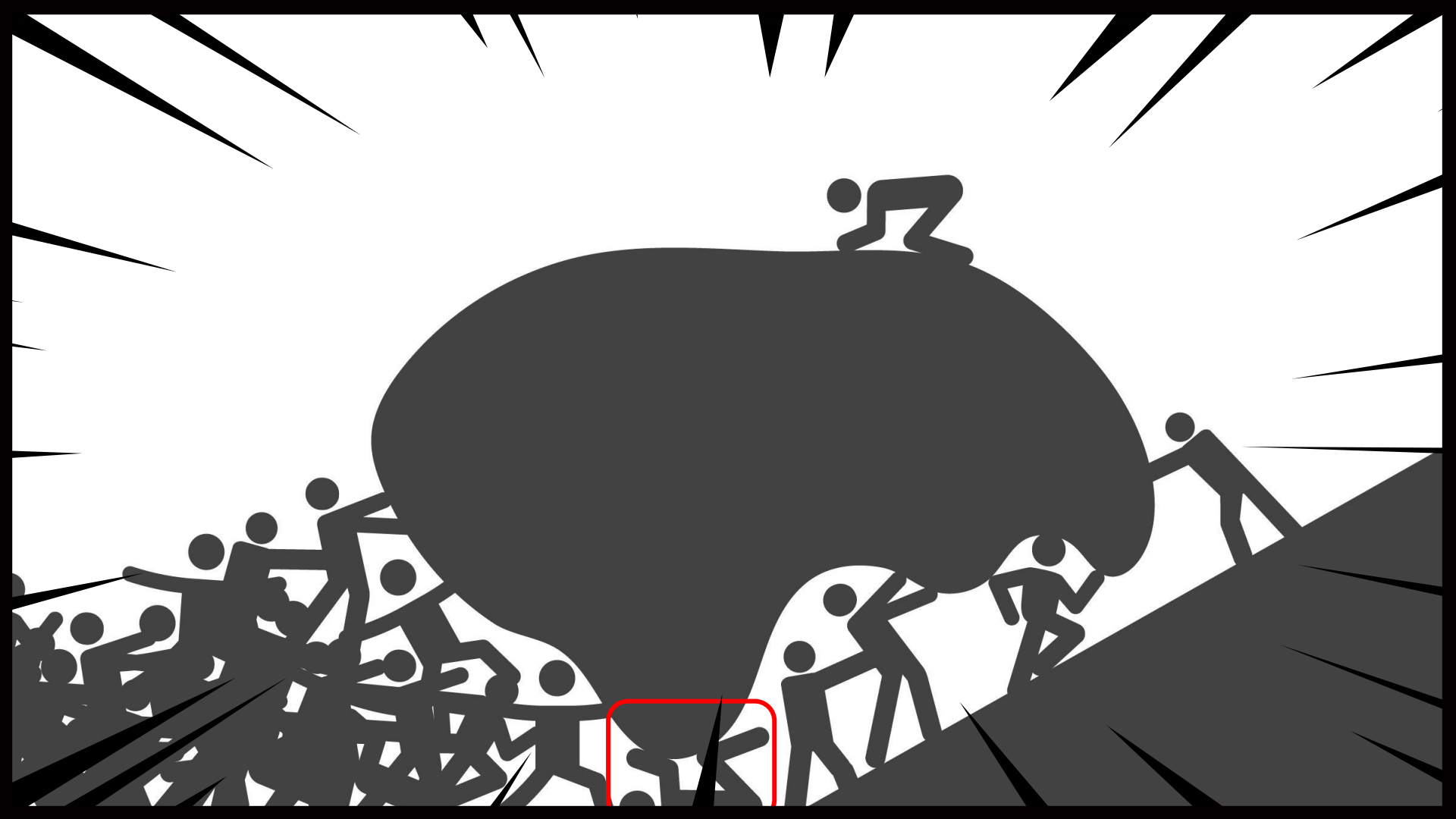
RED HAT DEVELOPER

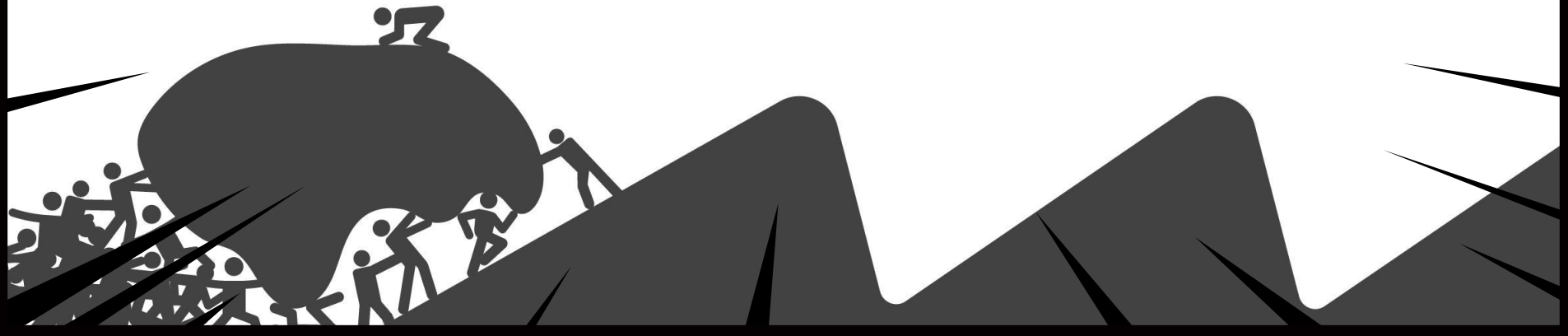

DEV
WALL OF CONFUSION
OPS

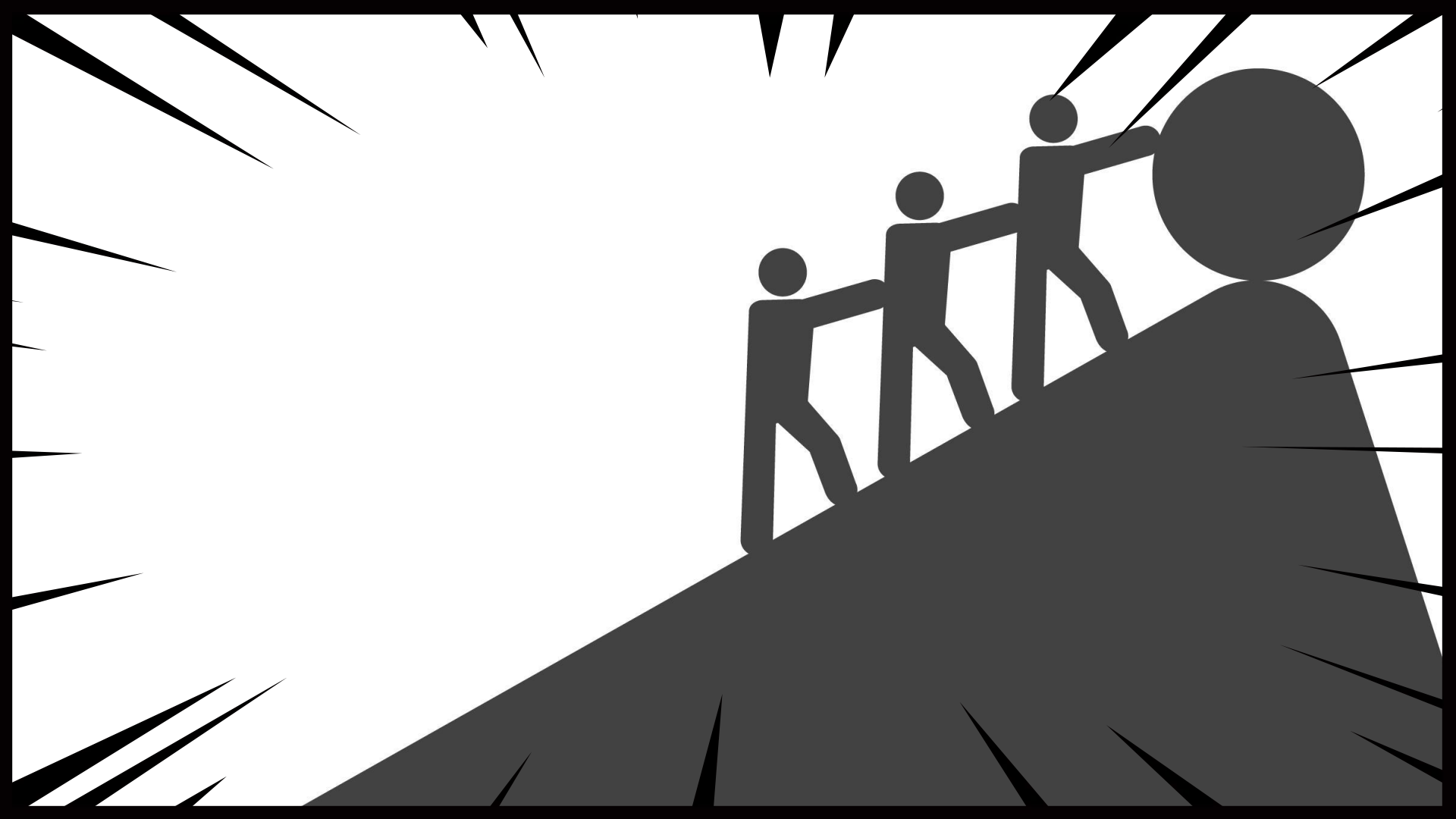

DEV
WALL OF CONFUSION
OPS



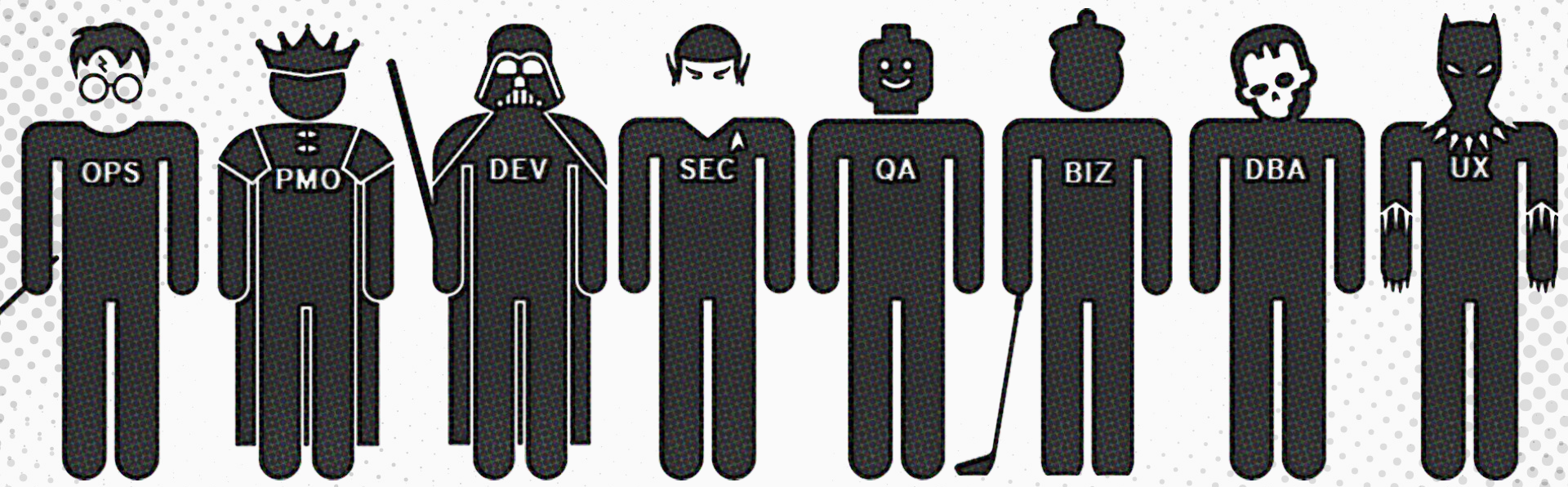


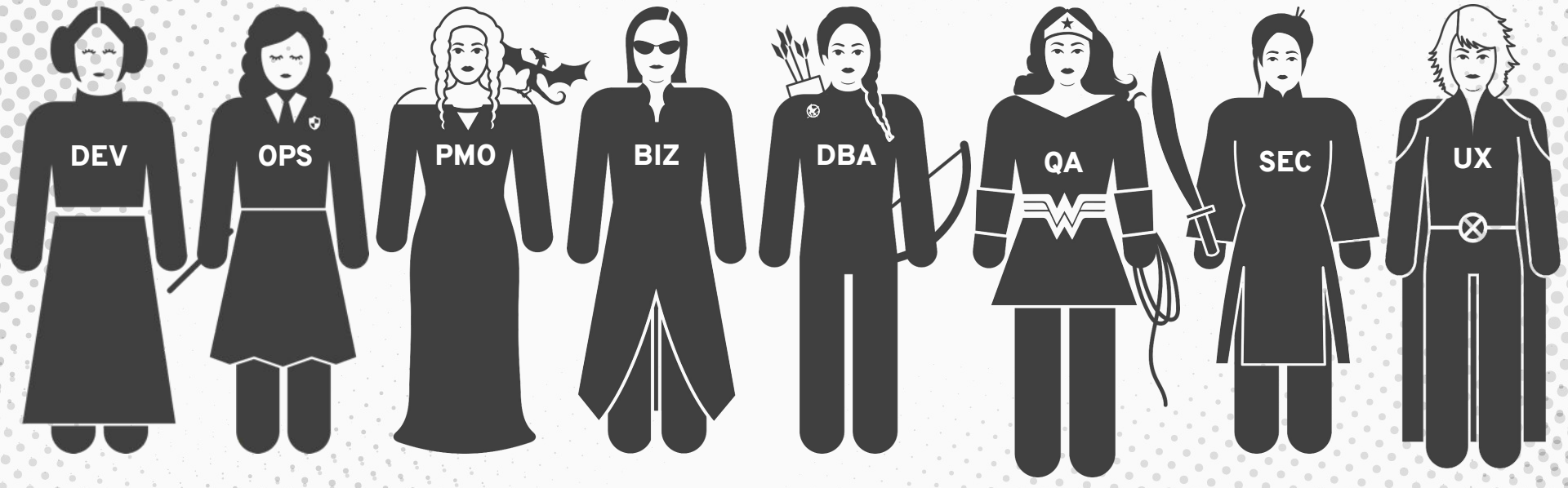


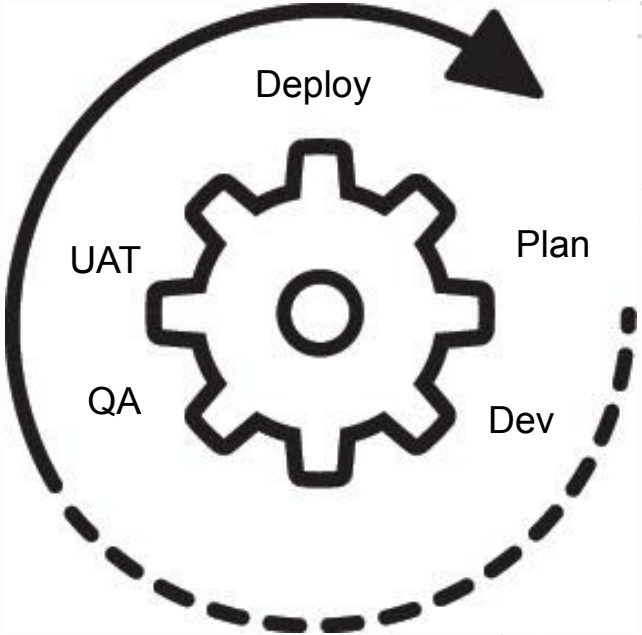




DEV
OPS
WALL OF CONFUSION

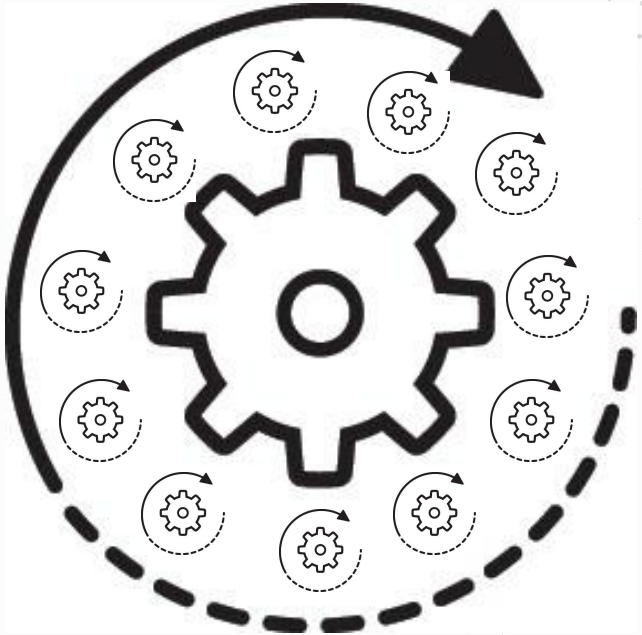

OPS
PMO
DEV
SEC
QA
BIZ
DBA
UX


DEV
OPS
PMO
BIZ
DBA
QA
SEC
UX

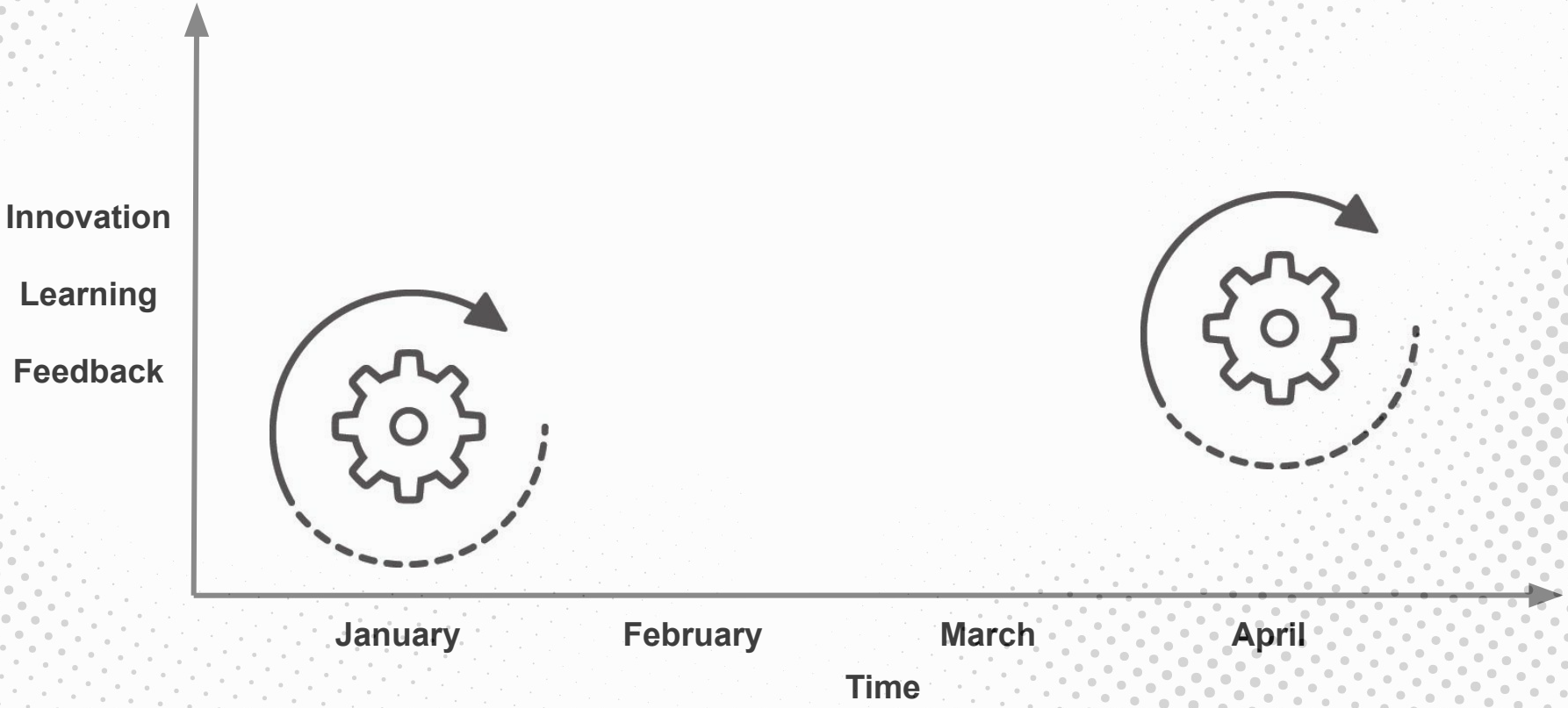


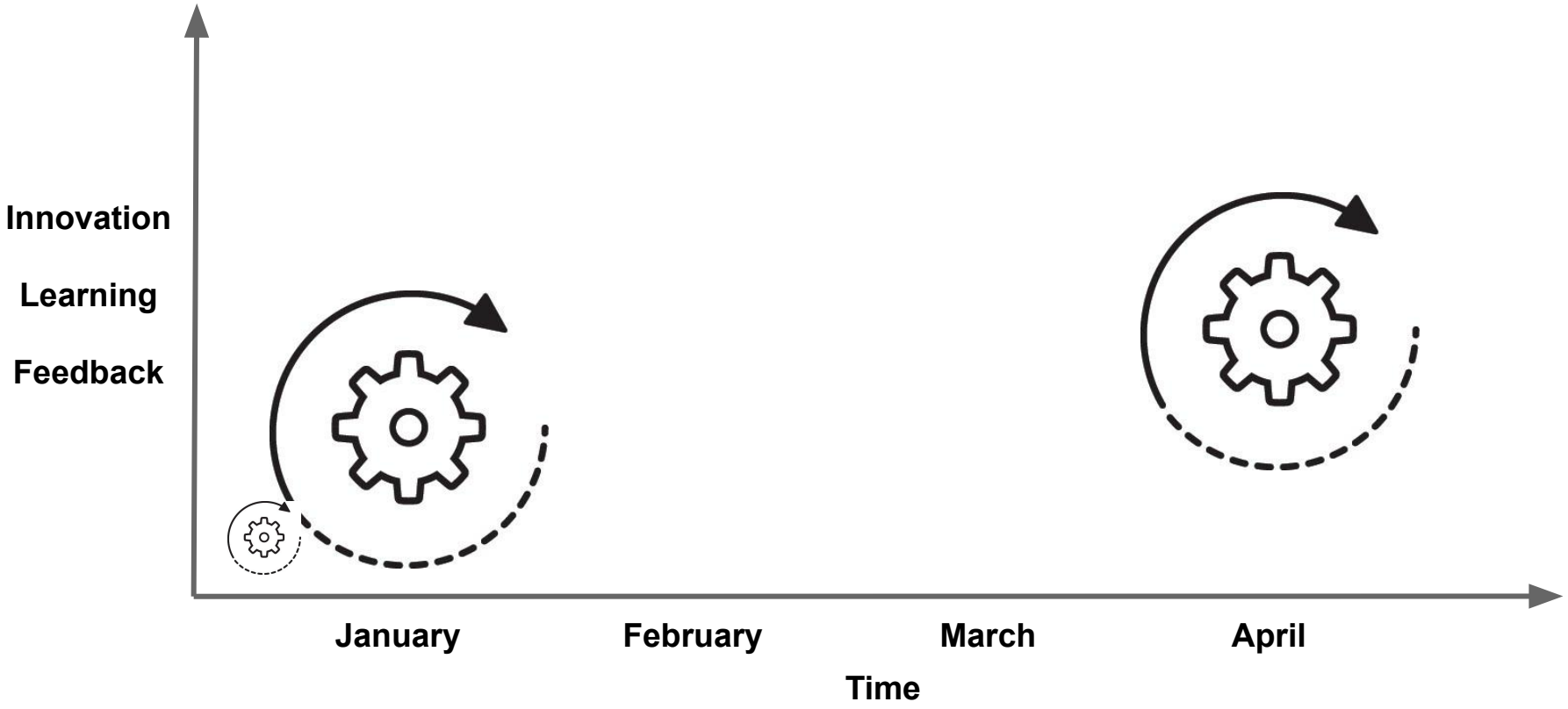
3 Month Deployment Cycle

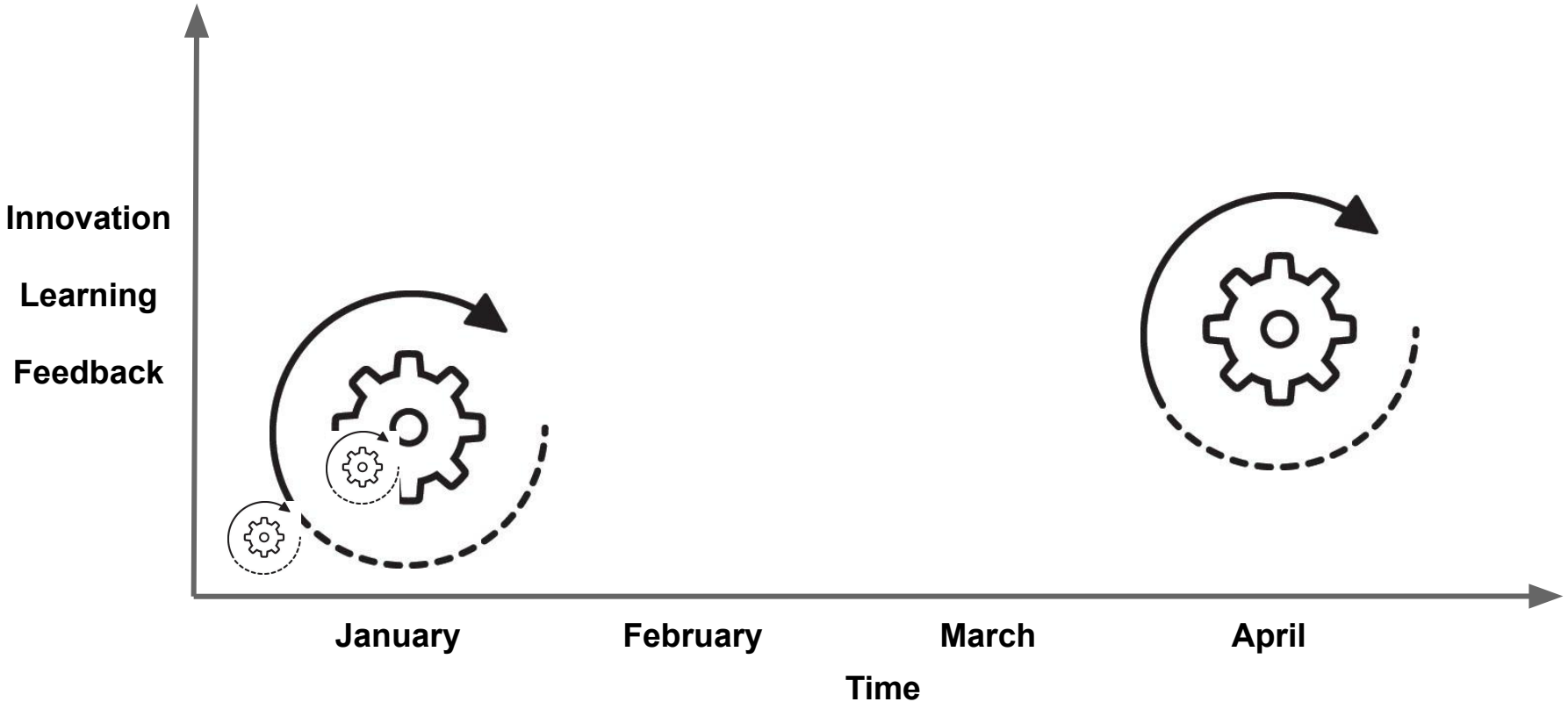
3 Months BEFORE you gain Feedback and Learn

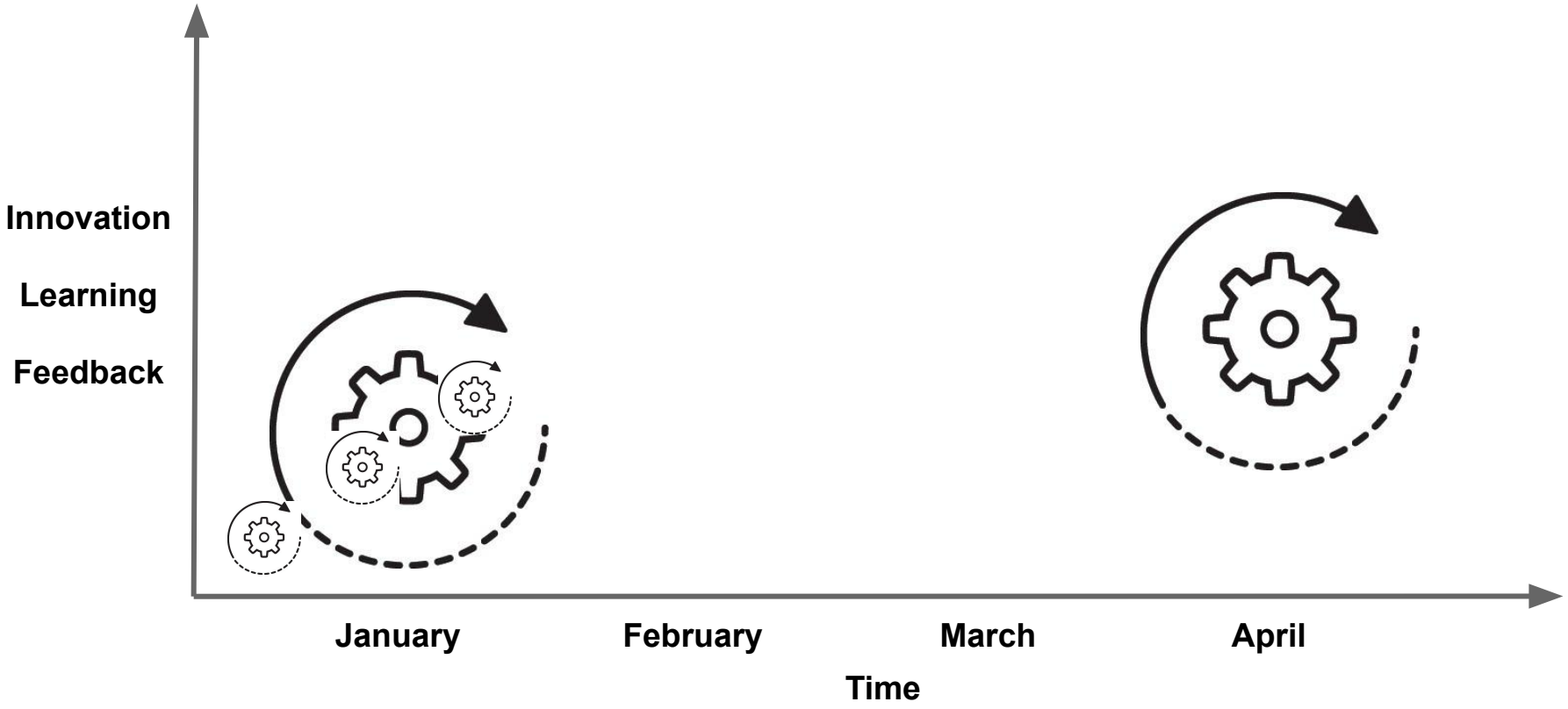


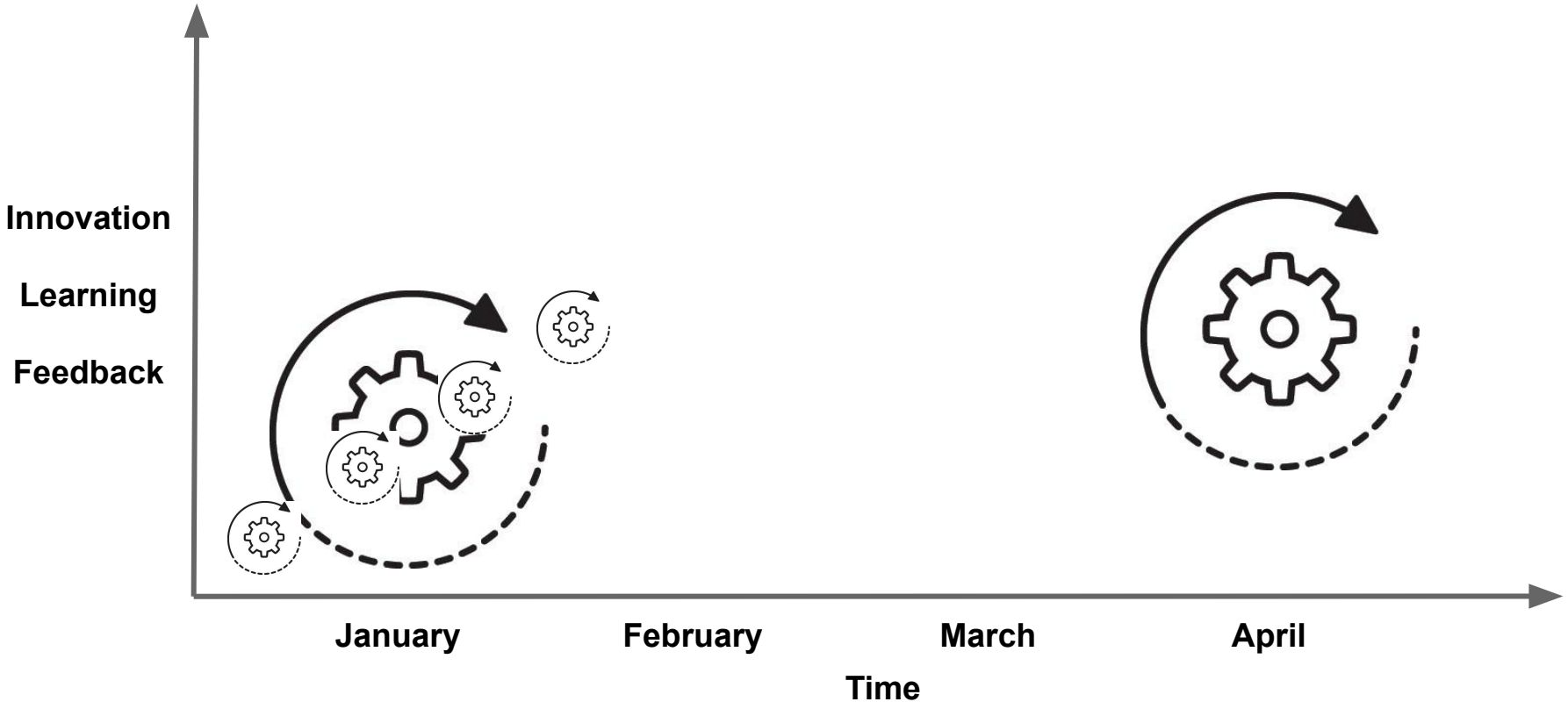
3 Months Versus 1 Week

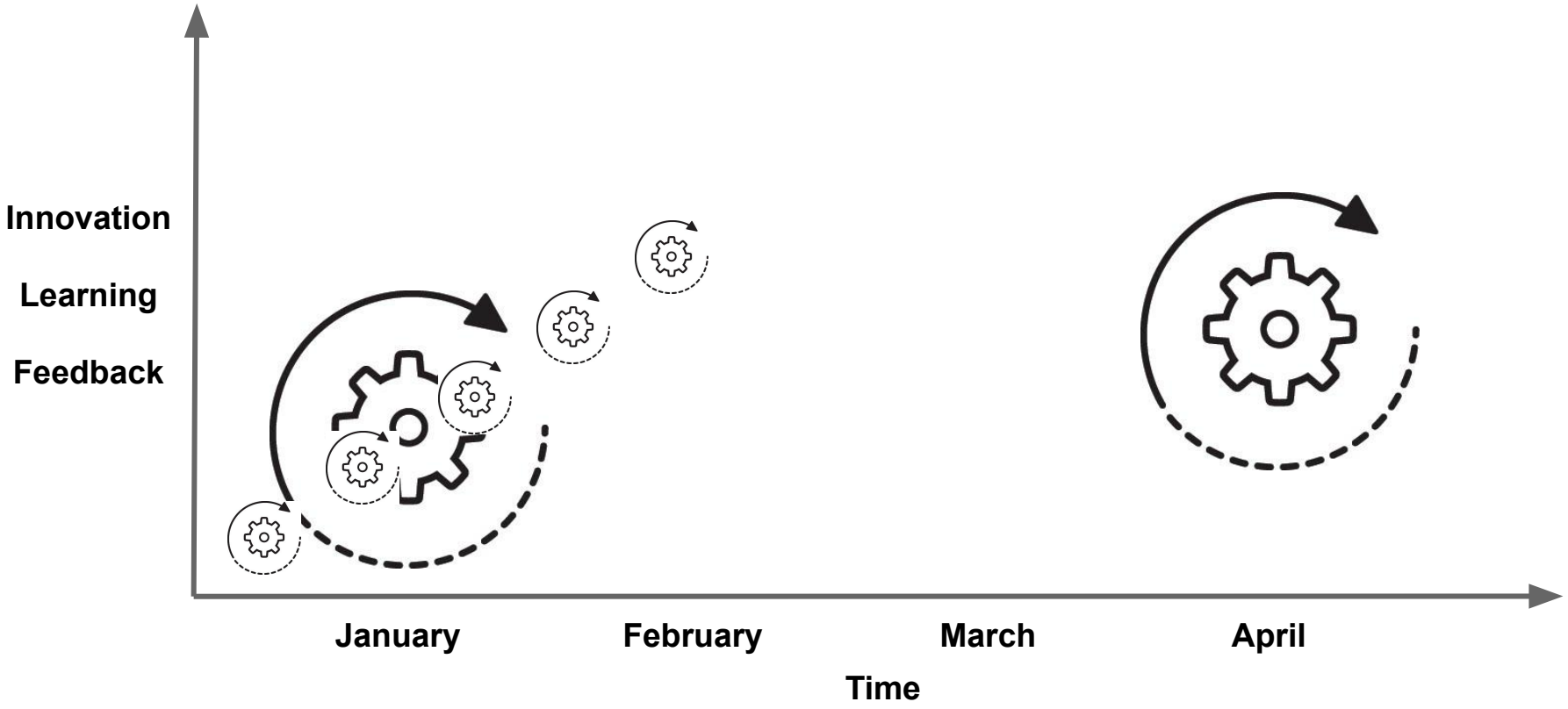

Innovation
Learning
Feedback
January
February
March
April
Time

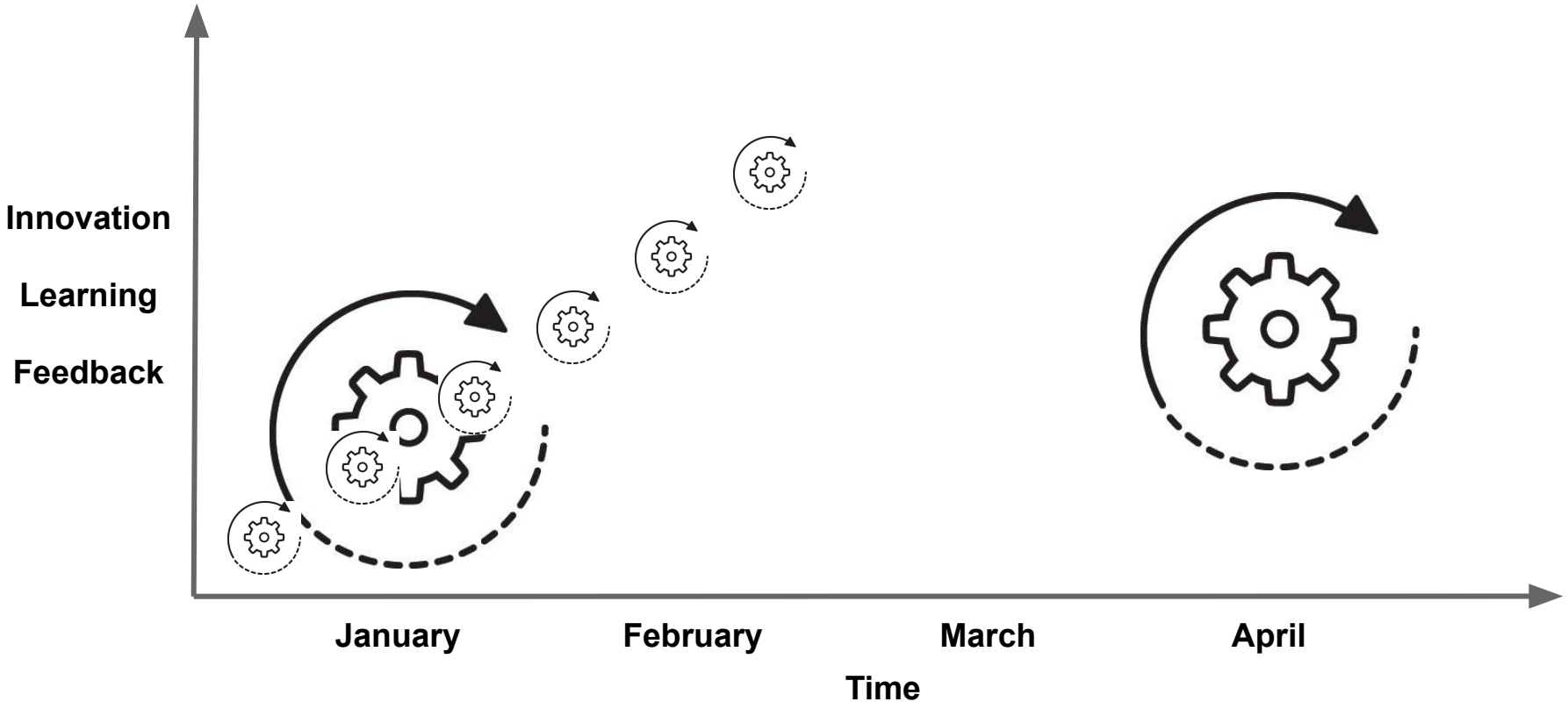

Innovation
Learning
Feedback
January
February
March
April
Time

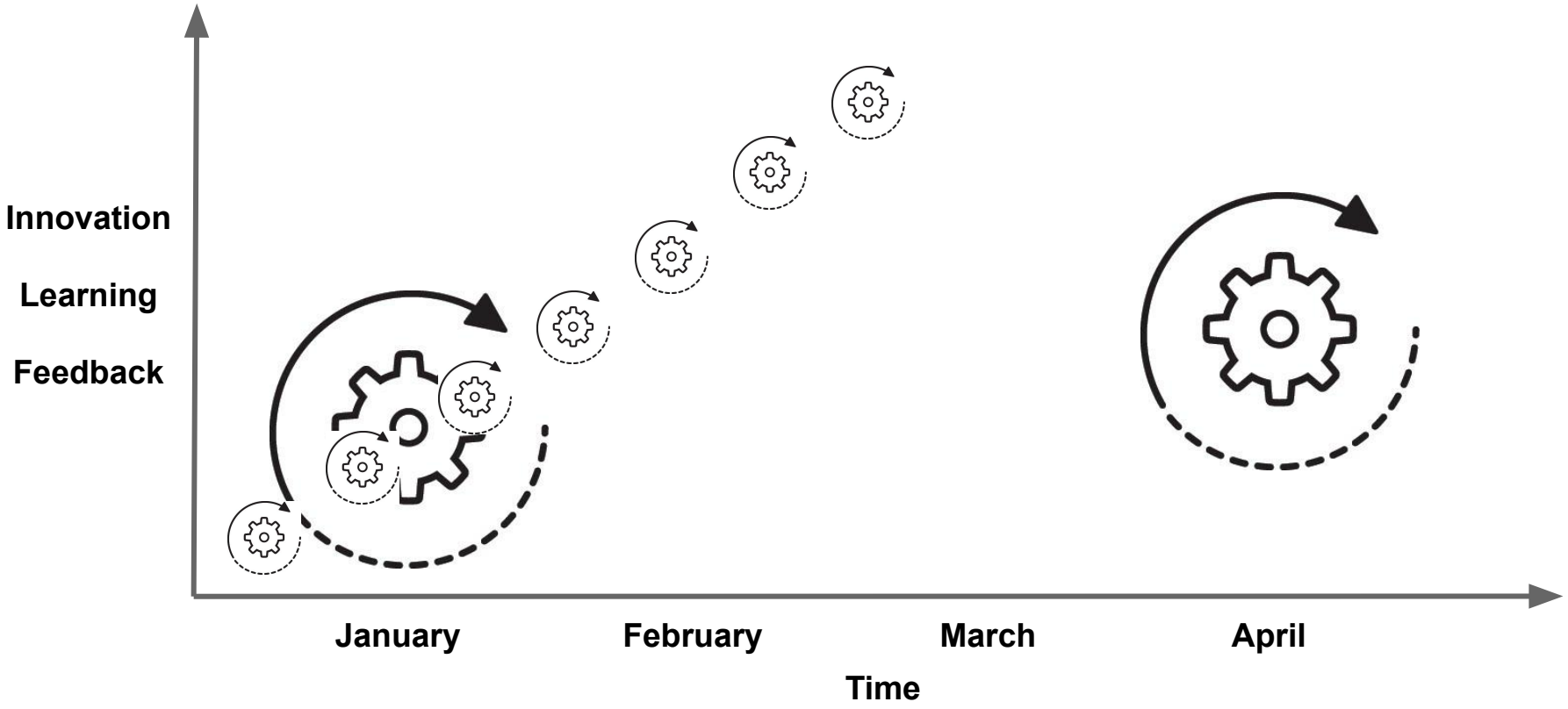

Innovation
Learning
Feedback
January
February
March
April
Time

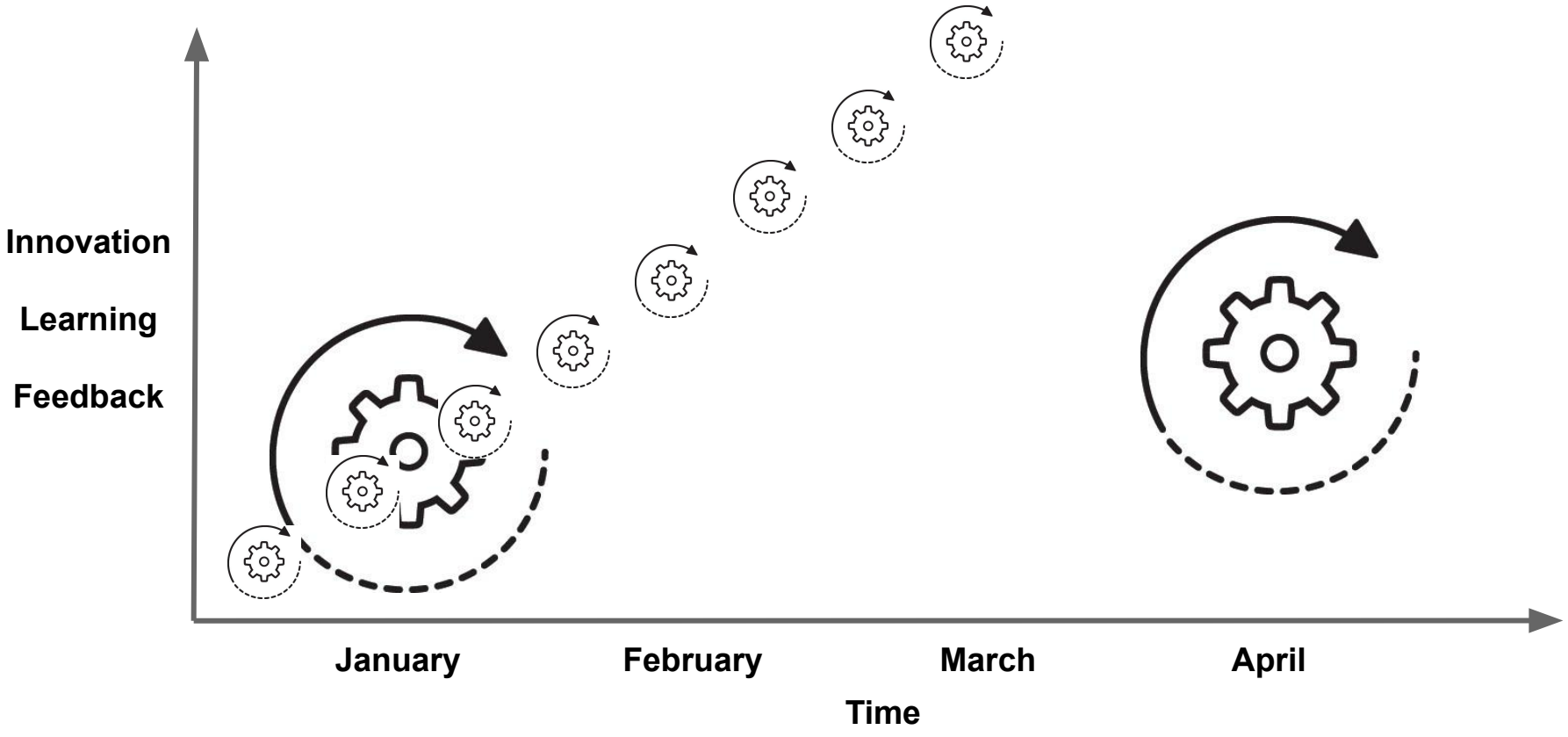

Innovation
Learning
Feedback
January
February
March
April
Time

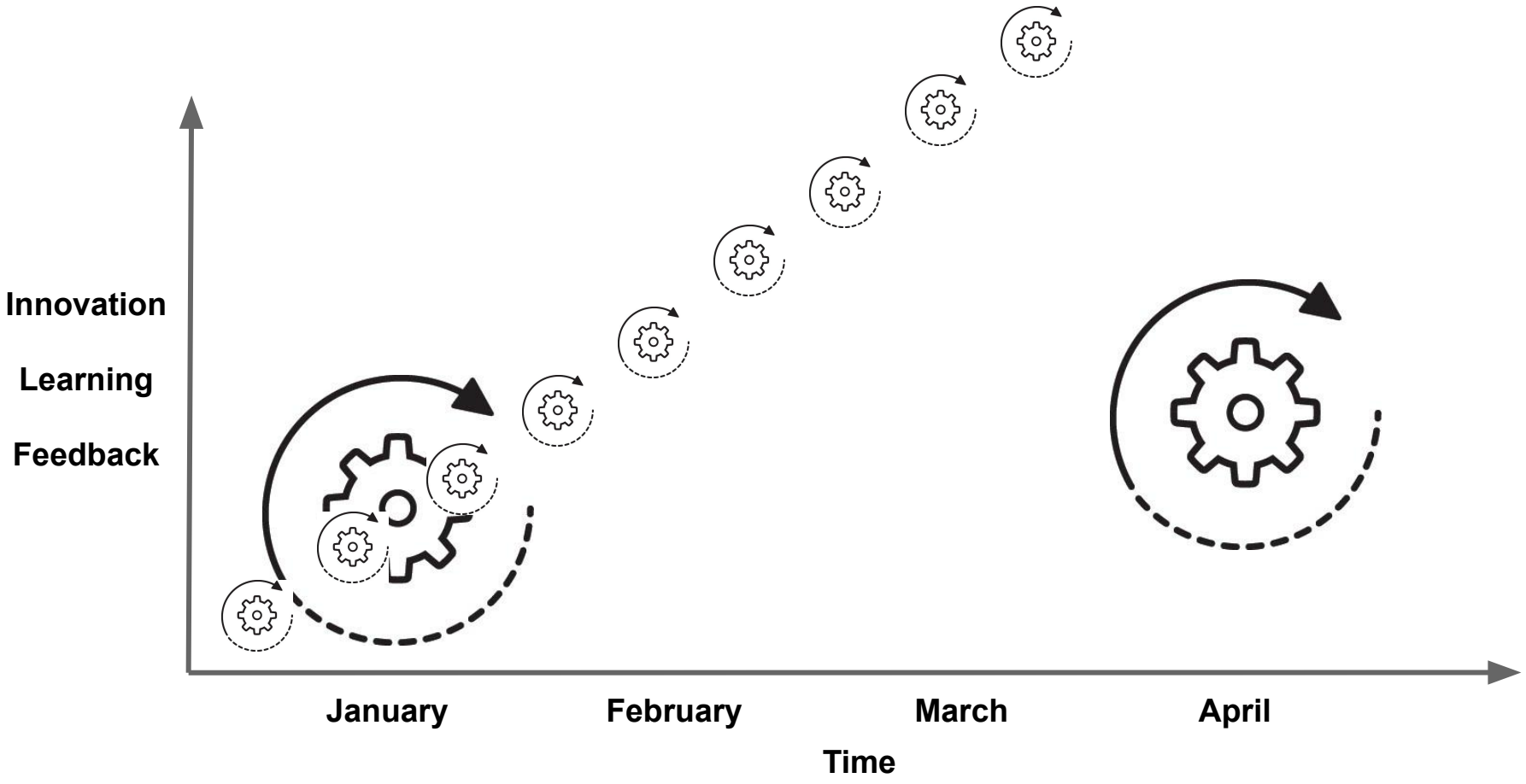

Innovation
Learning
Feedback
January
February
March
April
Time

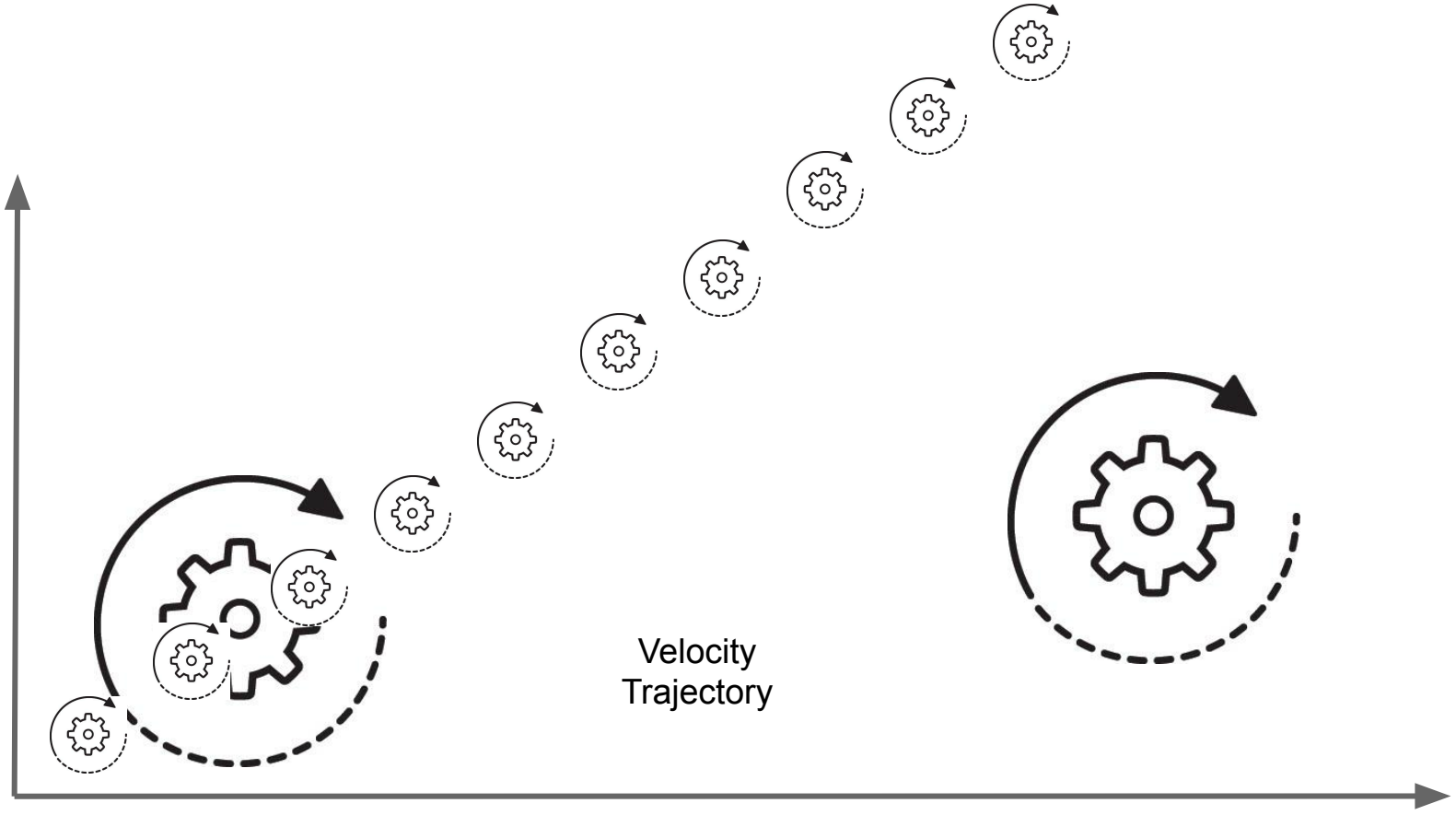

Innovation
Learning
Feedback
January
February
March
April
Time

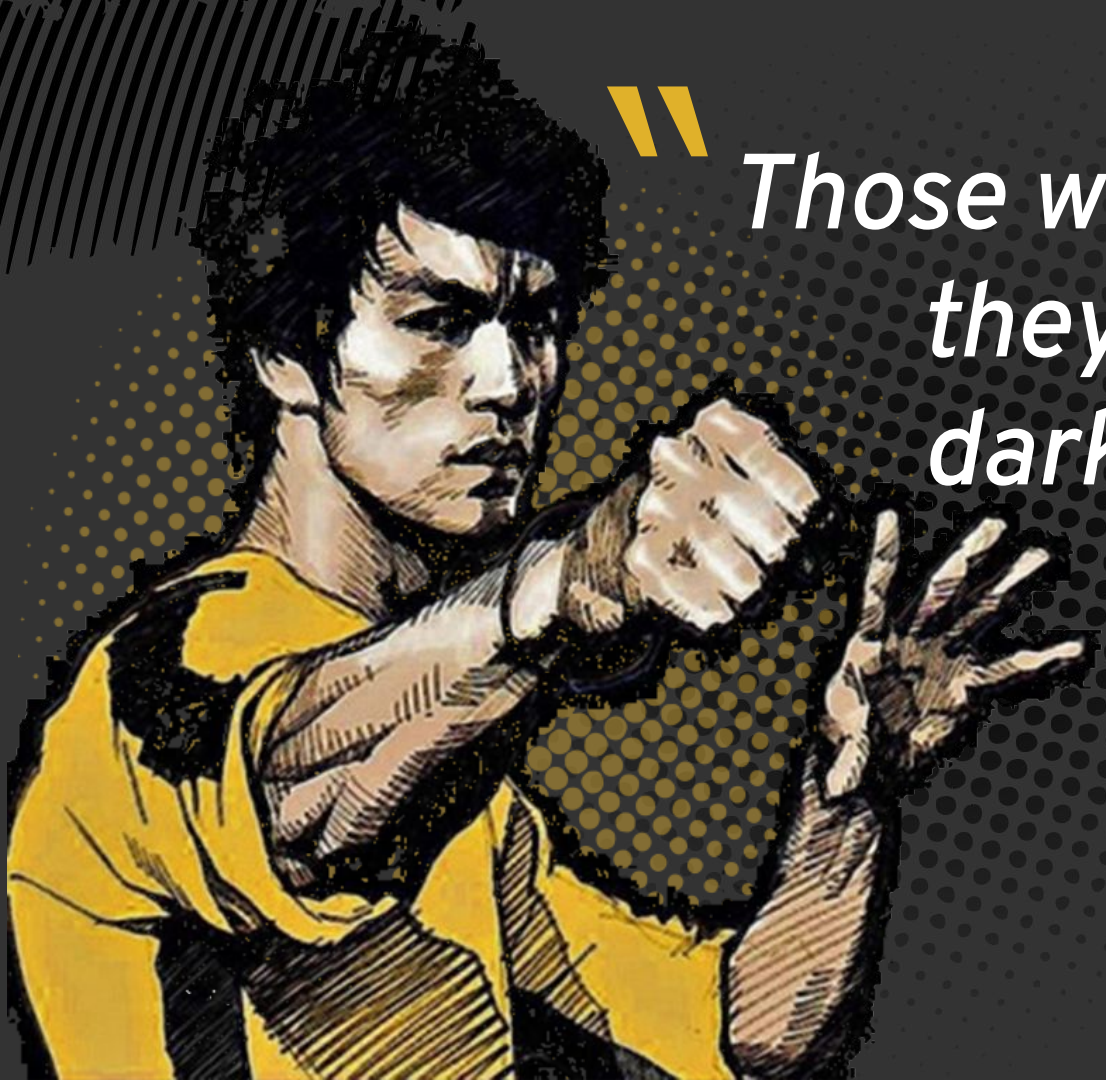

Innovation
Learning
Feedback
January
February
March
April
Time


Innovation
Learning
Feedback
January
February
March
April
Time


Innovation
Learning
Feedback
January
February
March
April
Time


Innovation
Learning
Feedback
January
February
March
April
Time


Velocity
Trajectory

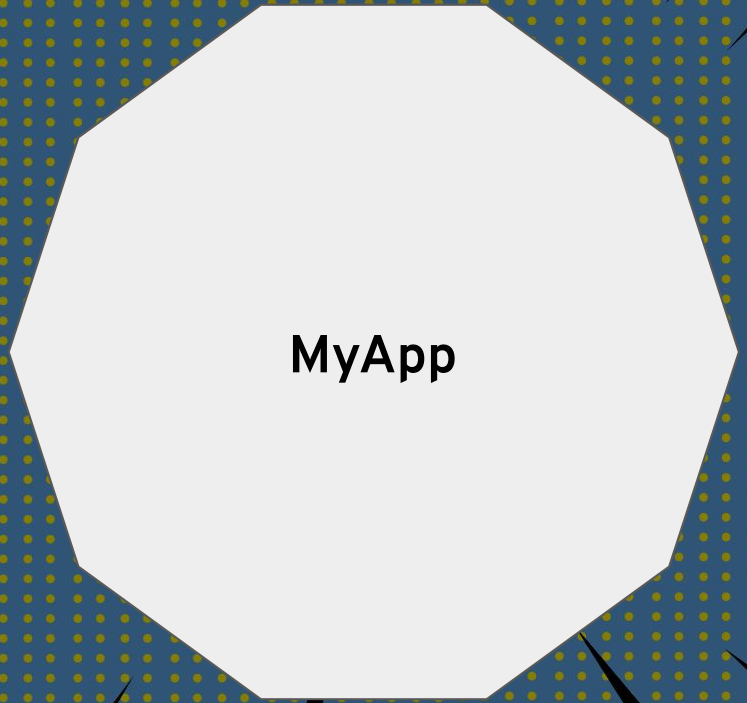


Those who are unaware they are walking in darkness will never seek the light.

— *Bruce Lee*

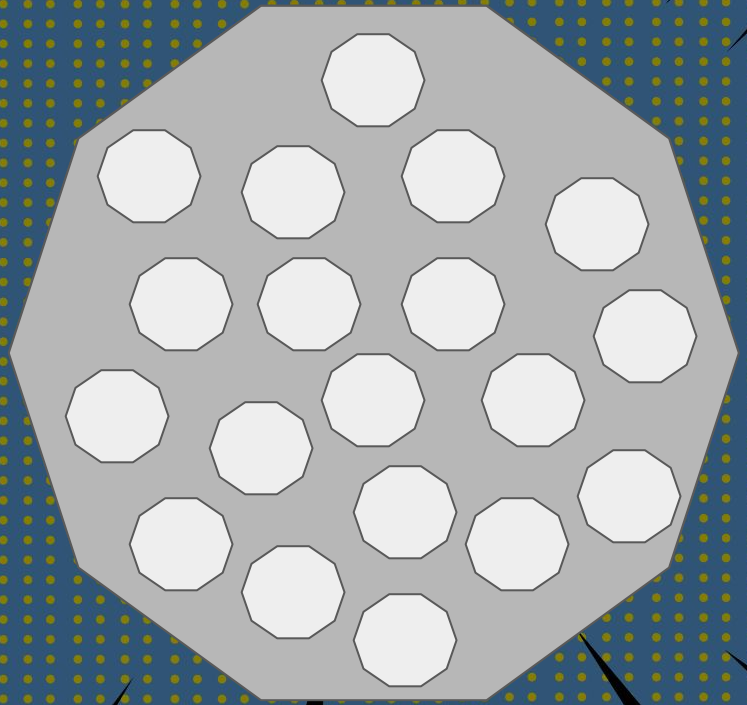
RED HAT DEVELOPER

Monolith



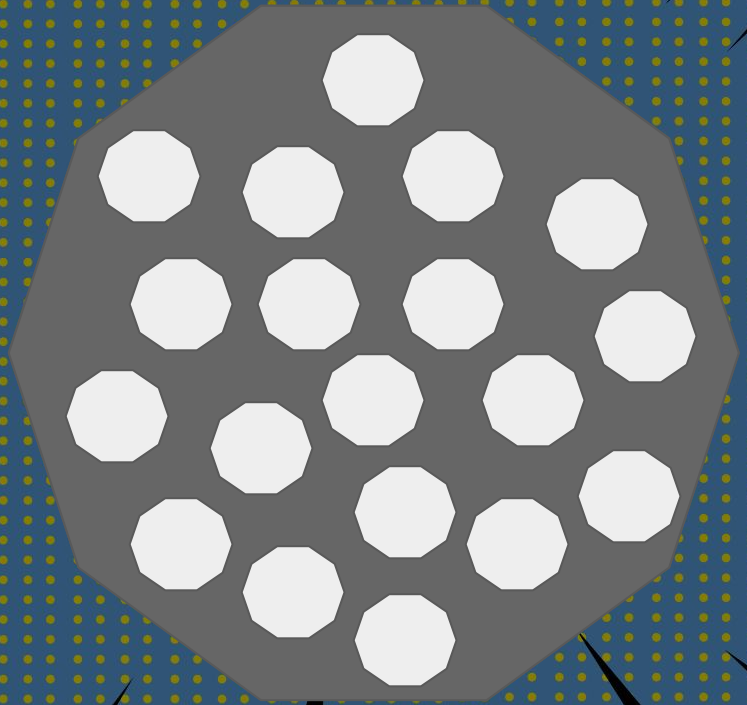
RED HAT DEVELOPER

The Application



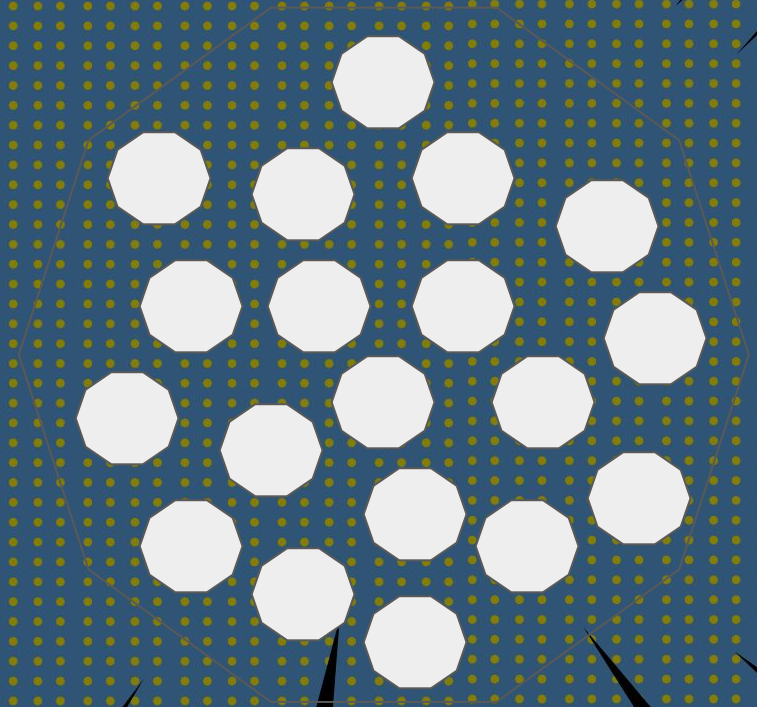
RED HAT DEVELOPER

Modules



RED HAT DEVELOPER

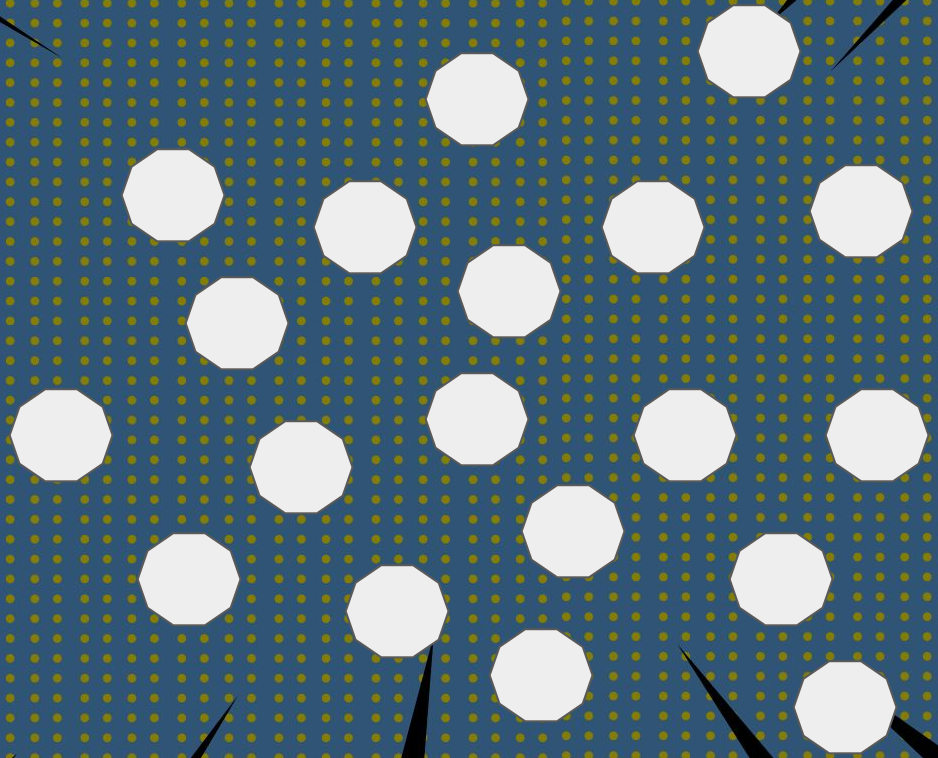
Microservices



@burrsutter

RED HAT DEVELOPER

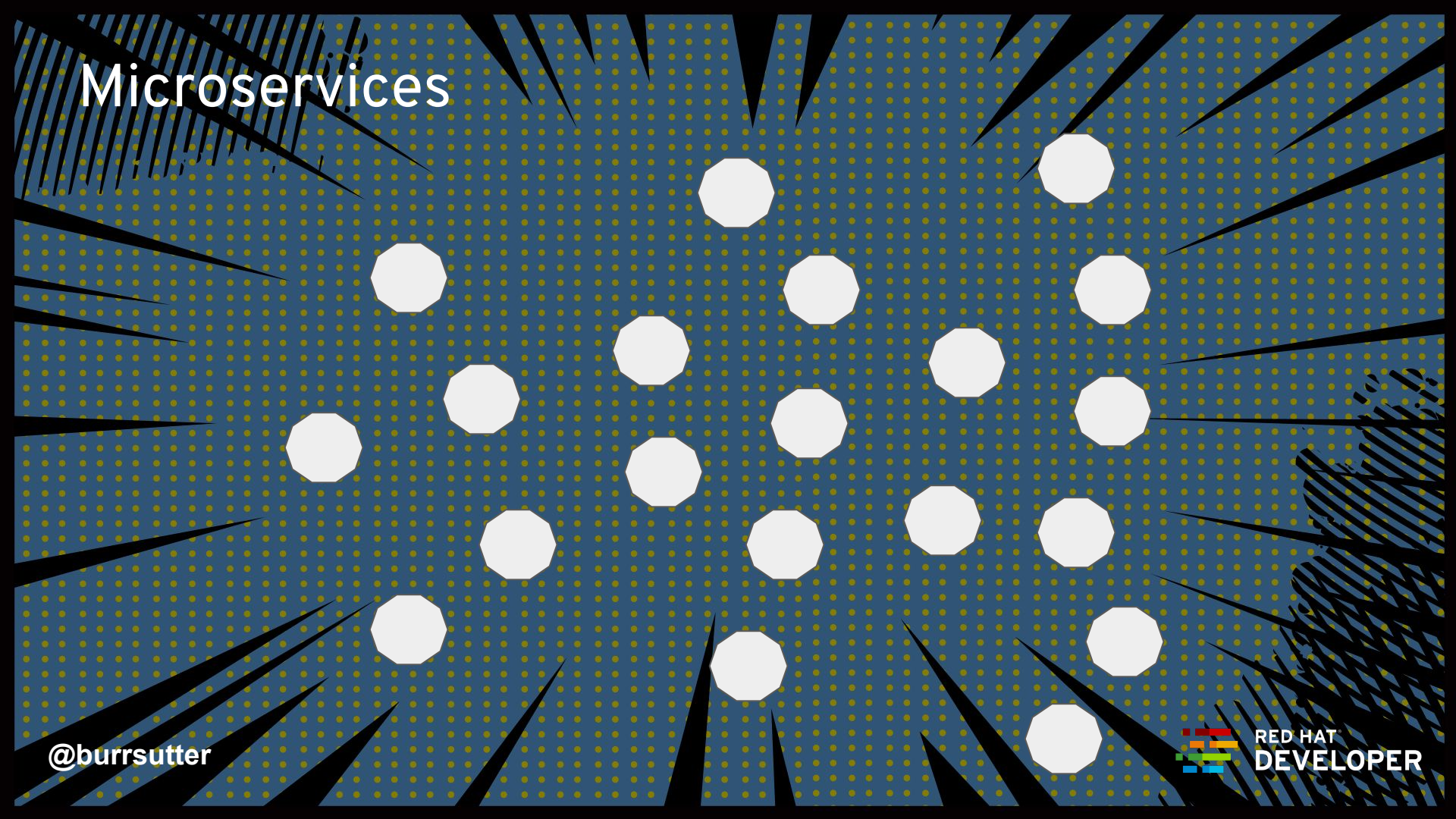
Microservices



@burrsutter

RED HAT DEVELOPER

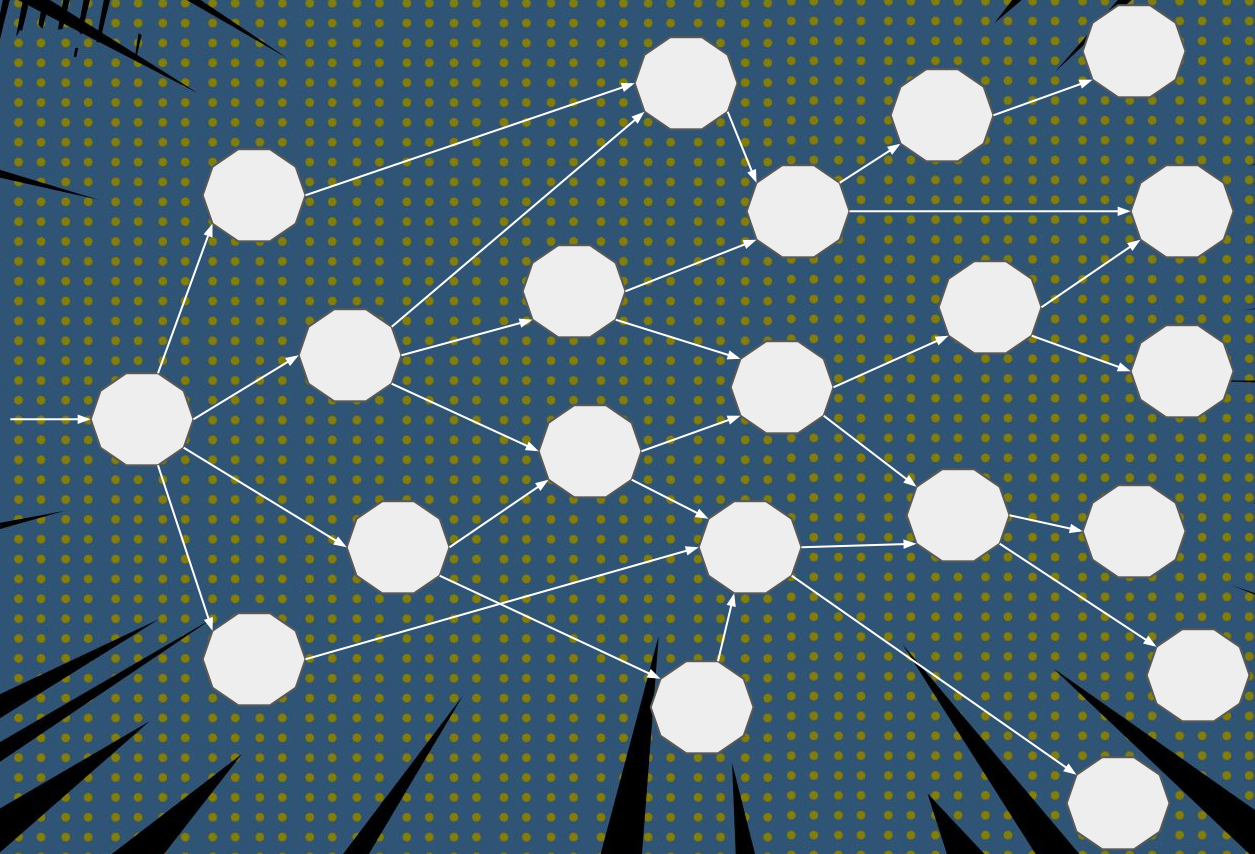
Microservices



@burrsutter

RED HAT DEVELOPER

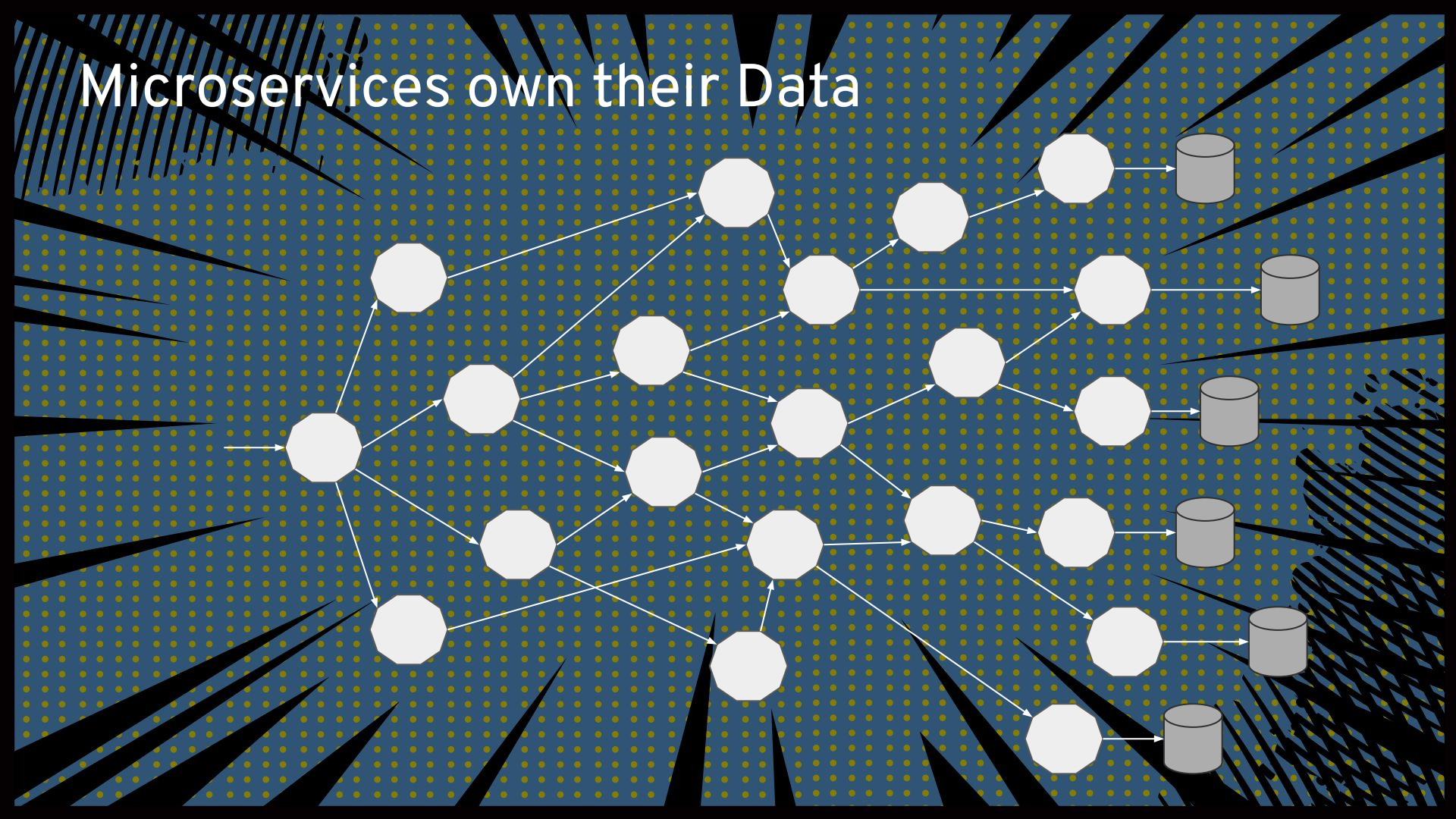
Network of Services



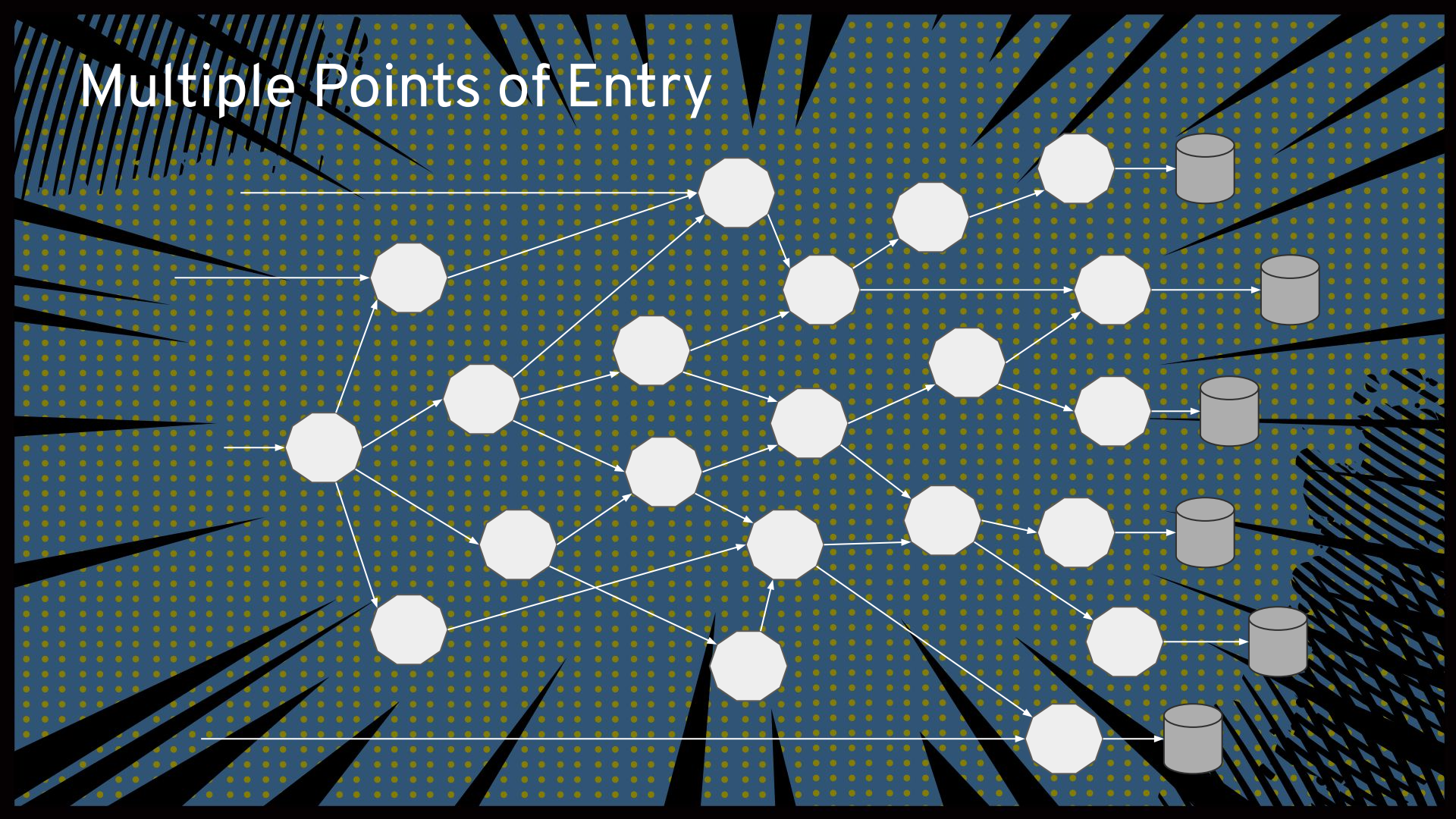
@burrsutter

RED HAT DEVELOPER

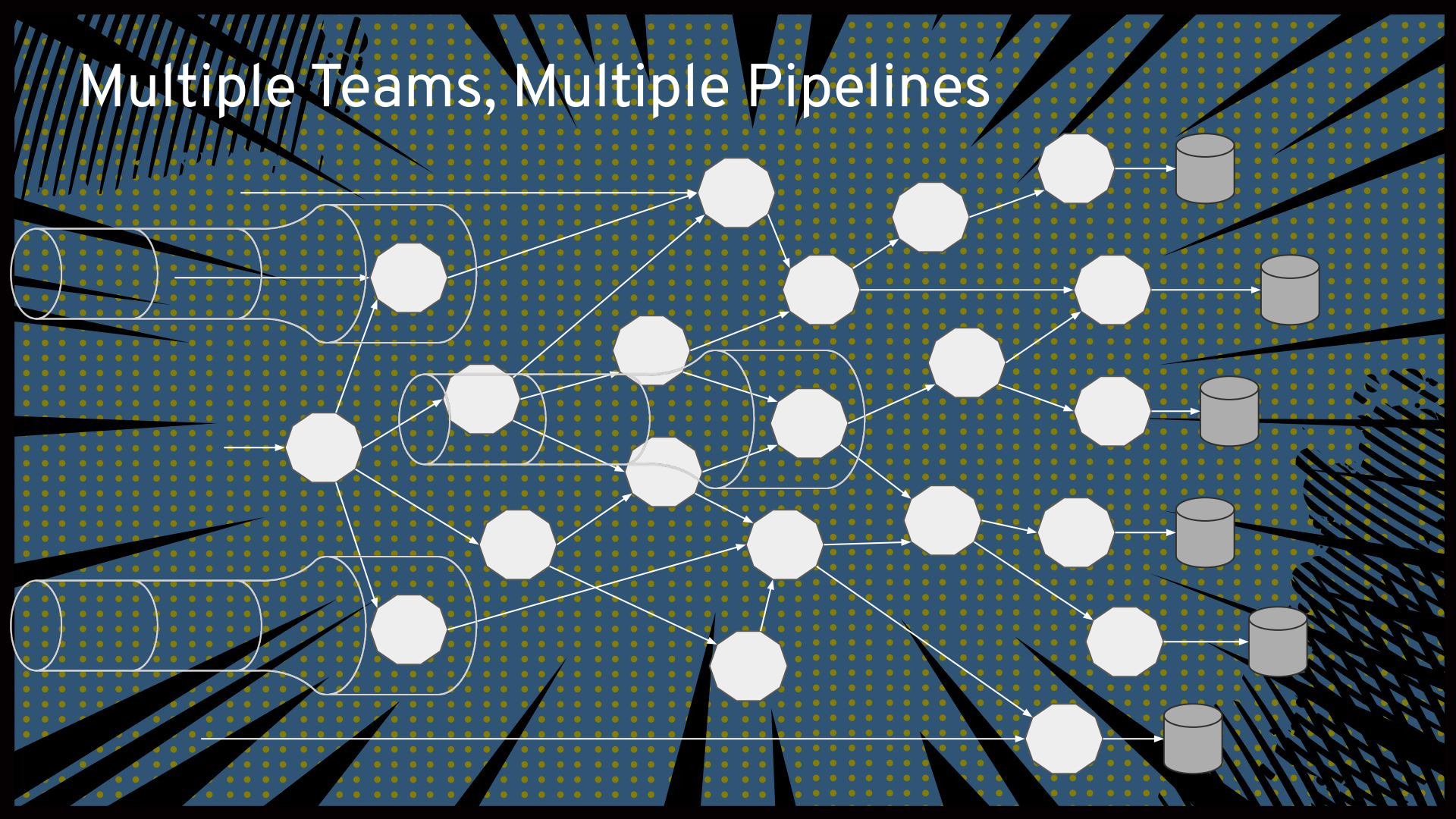
Microservices own their Data



Multiple Points of Entry

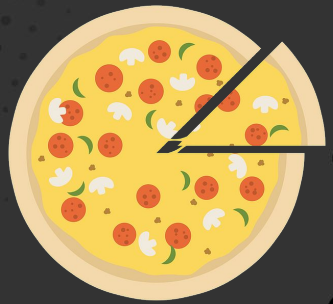


Multiple Teams, Multiple Pipelines



Microservices Principles

1. Deployment **Independence** - updates to an individual microservice have no negative impact to any other component of the system. Optimized for **Replacement**
2. Organized around **business** capabilities
3. **Products** not Projects
4. **API** Focused
5. **Smart** endpoints and dumb pipes
6. Decentralized Governance
7. Decentralized Data Management
8. Infrastructure Automation (infrastructure as code)
9. Design for failure
10. Evolutionary Design



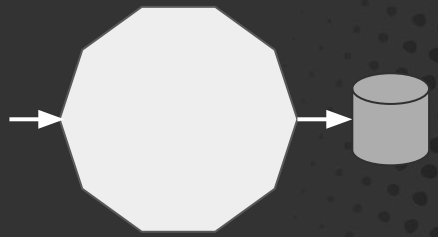
2 Pizza Team

@burrsutter

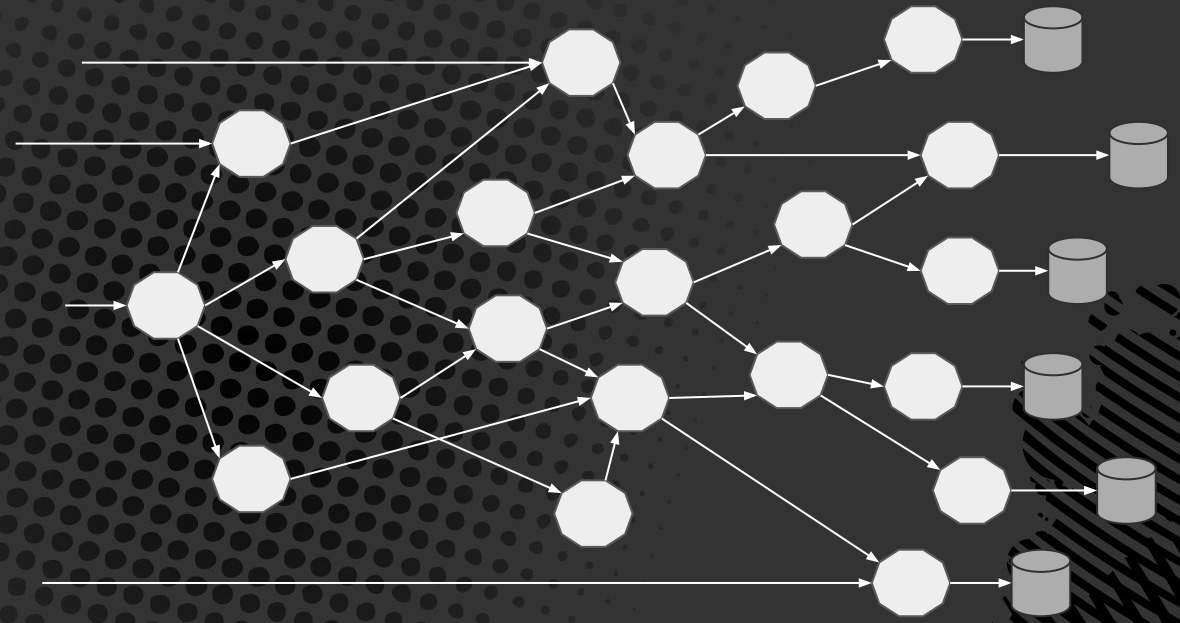
RED HAT DEVELOPER

Old School

New School



Love Thy Mono

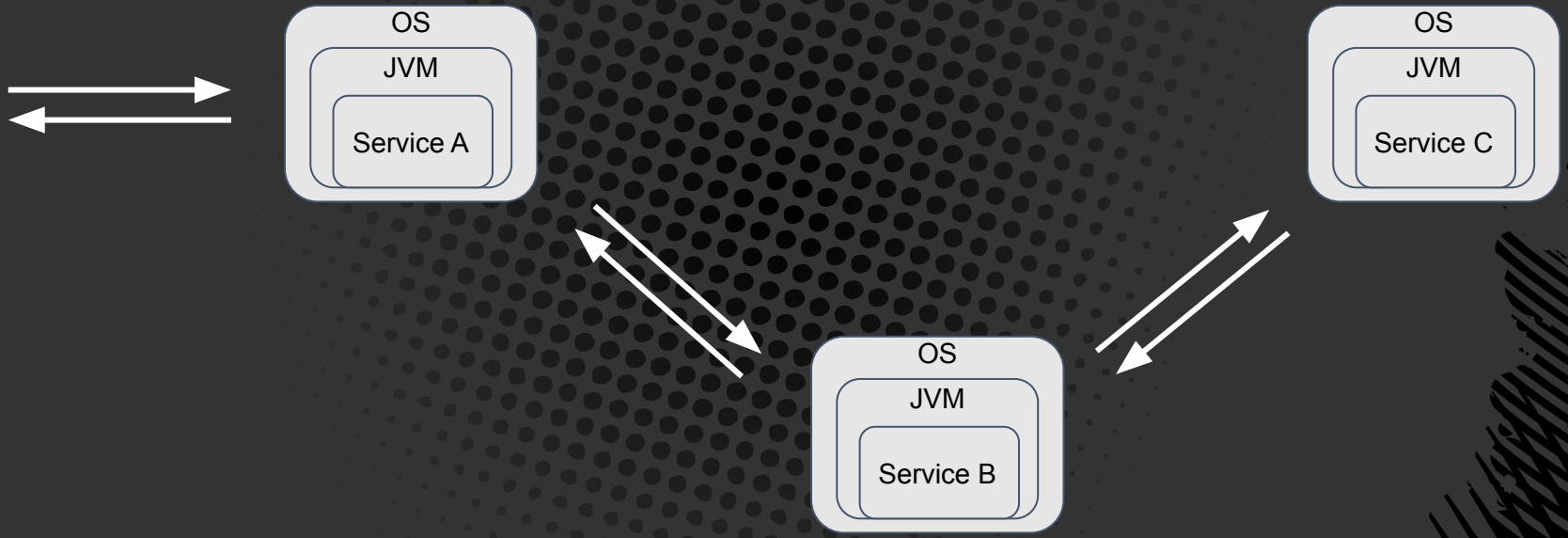


OPENSHIFT

RED HAT® DEVELOPER

@burrsutter

Microservices == Distributed Computing



RED HAT DEVELOPER

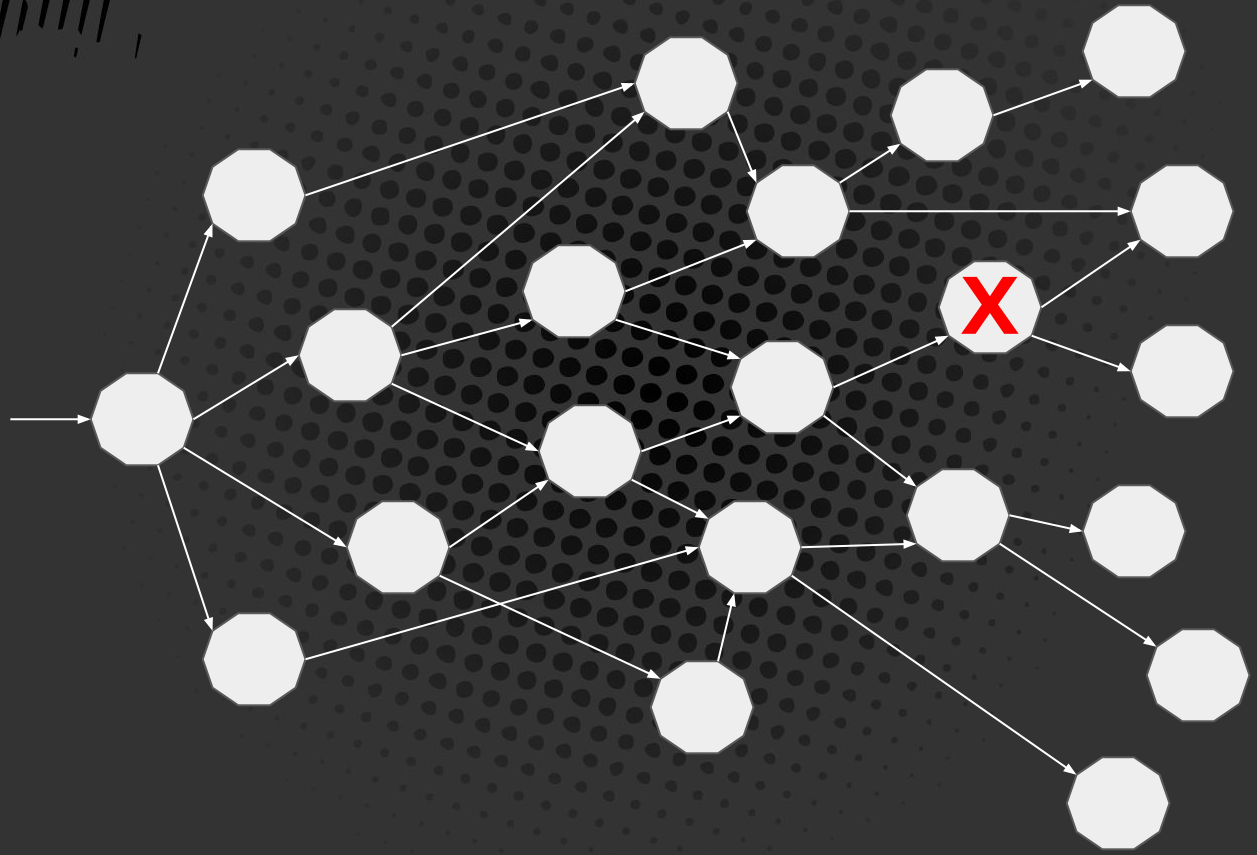
Fallacies of Distributed Computing

- The Network is Reliable
- Latency is zero
- Bandwidth is infinite
- Topology does not change
- There is one administrator
- Transport cost is zero
- The network is homogeneous

https://en.wikipedia.org/wiki/Fallacies_of_distributed_computing

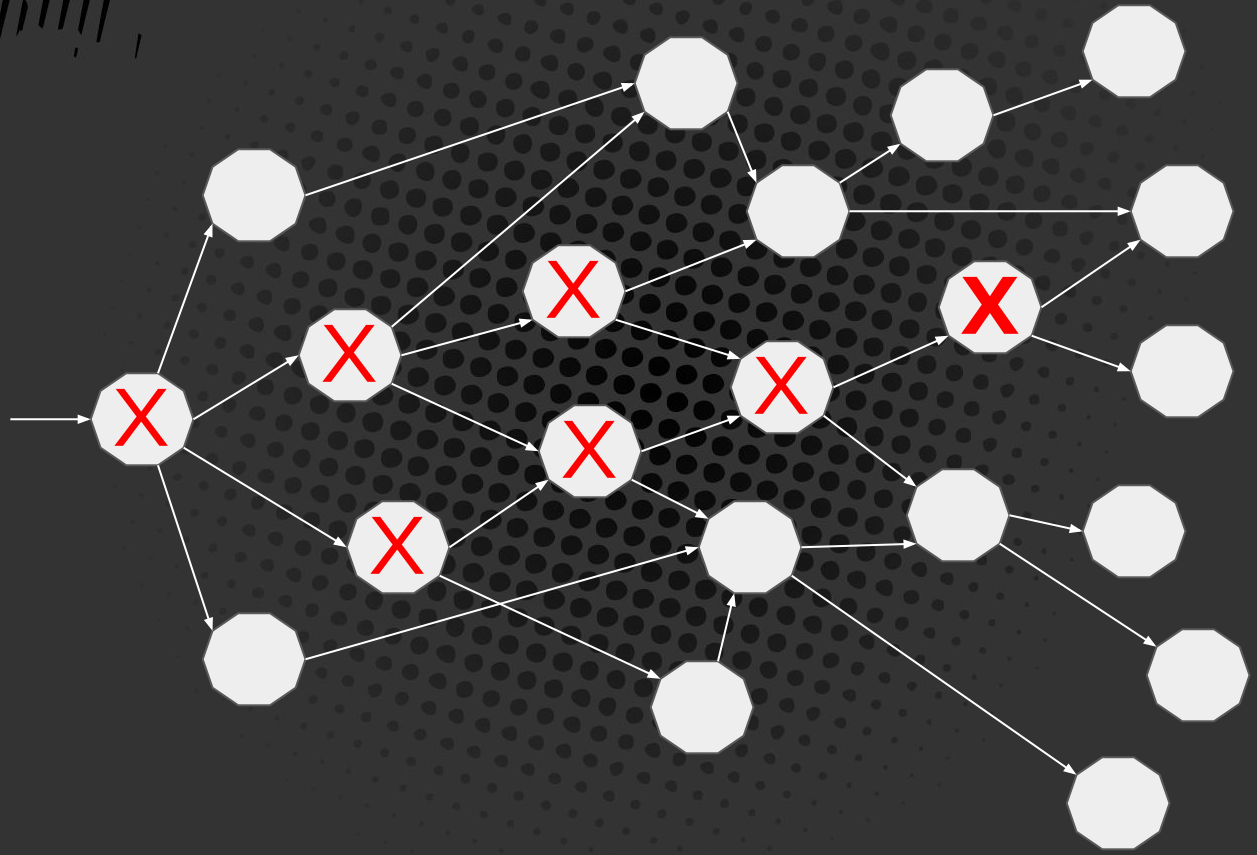
RED HAT DEVELOPER

Failure of a Service



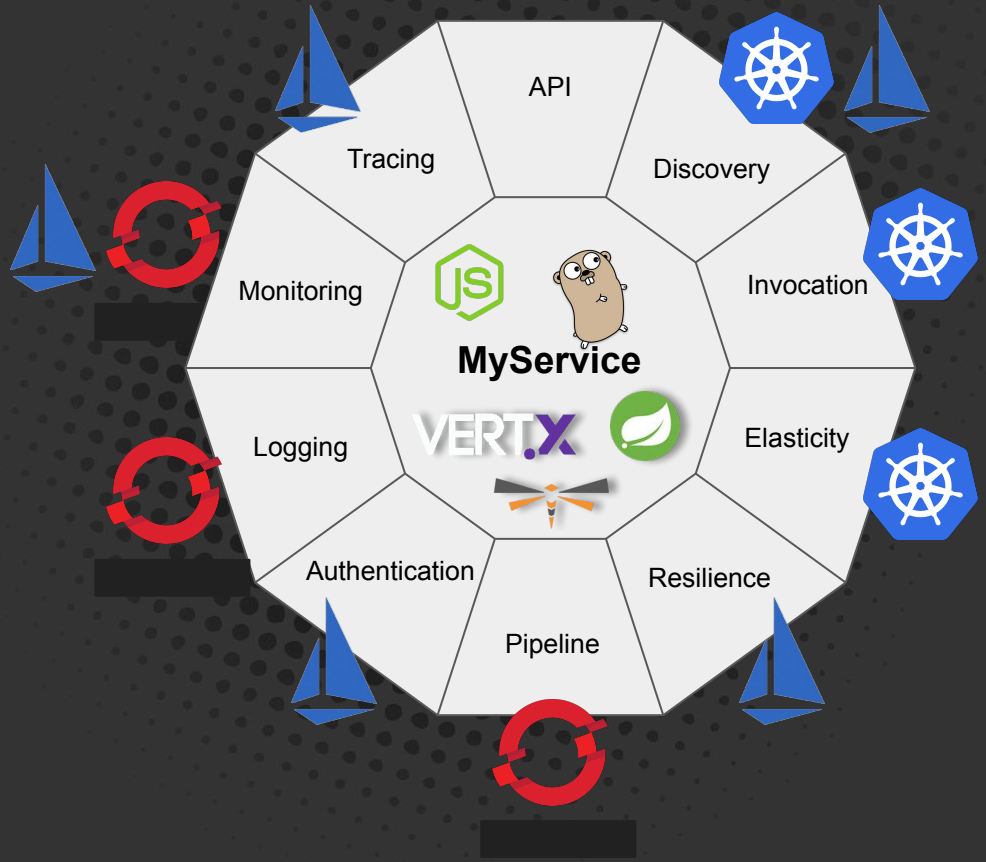
RED HAT DEVELOPER

Cascading Failure



RED HAT® DEVELOPER

Microservices'ilities + Istio

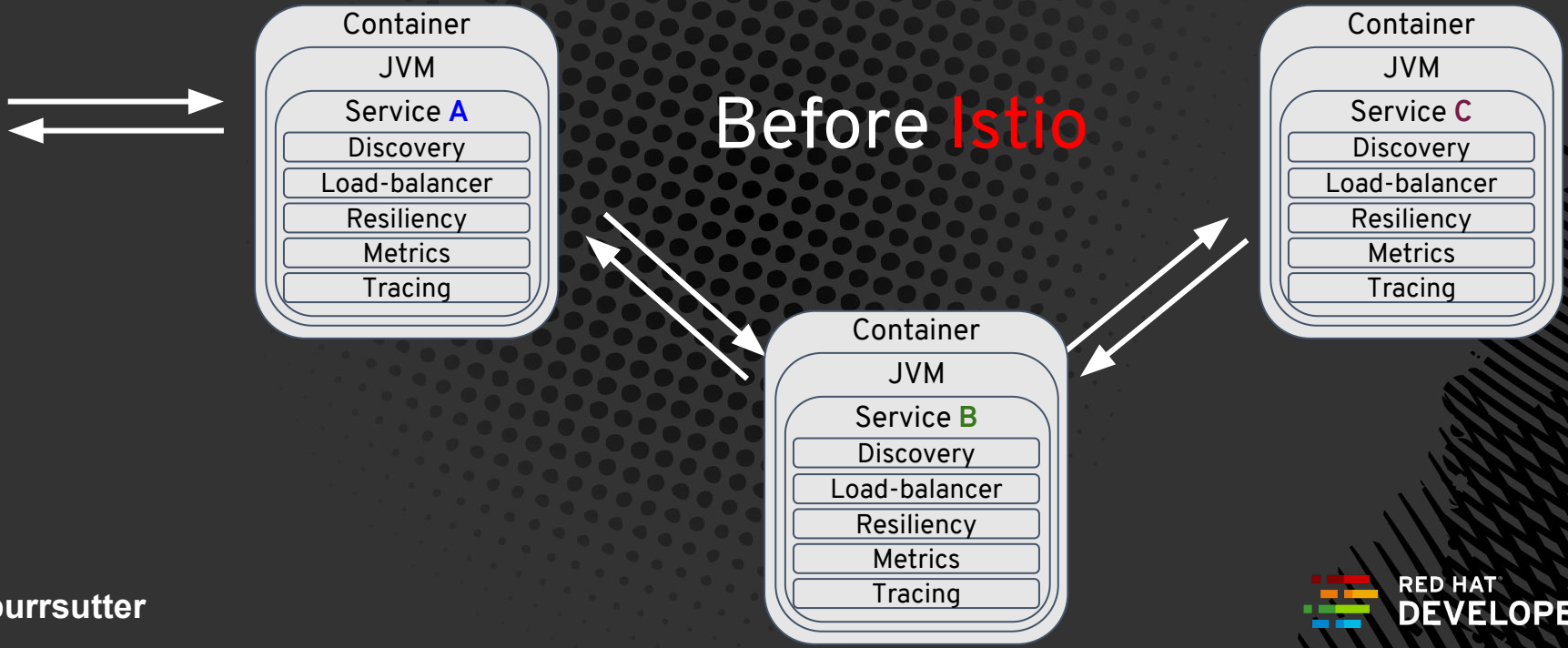


RED HAT DEVELOPER

Istio

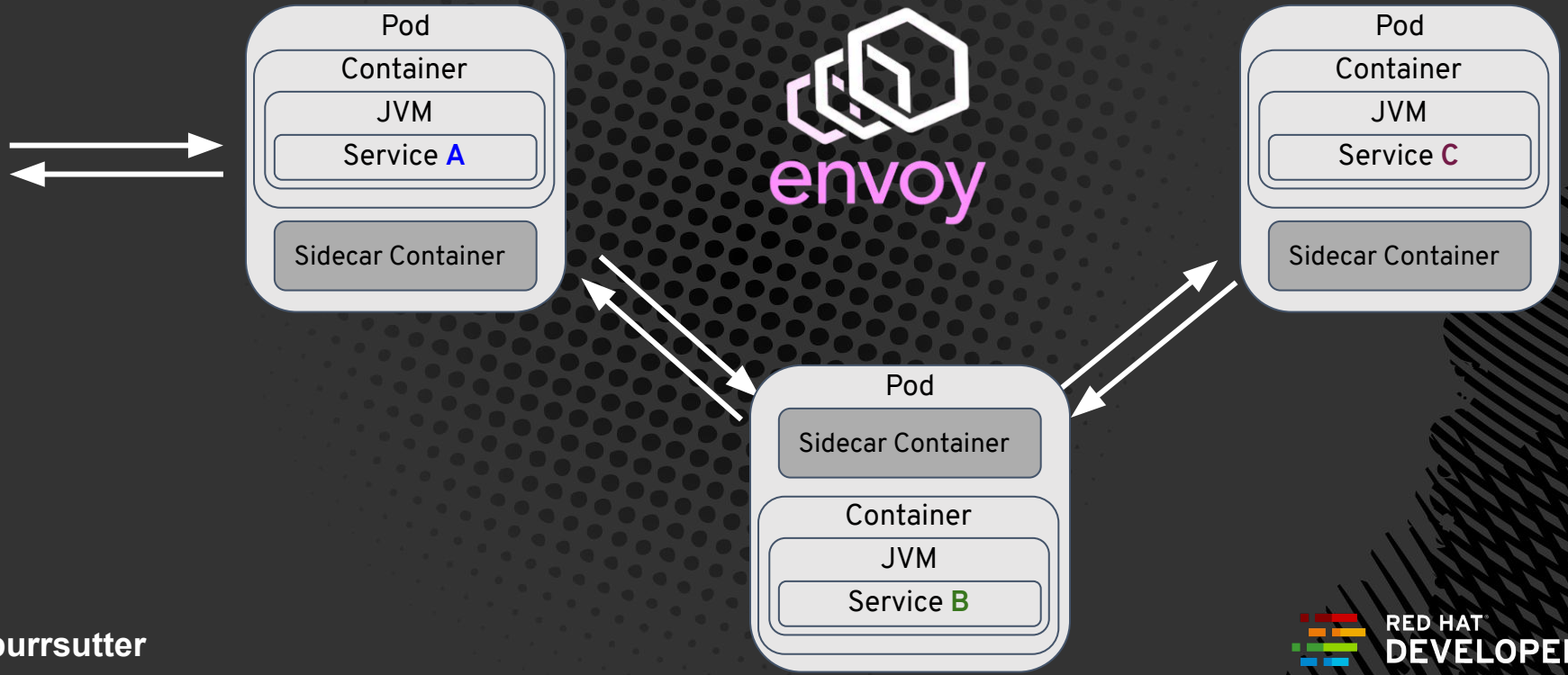
RED HAT®
DEVELOPER

Microservices embedding Capabilities



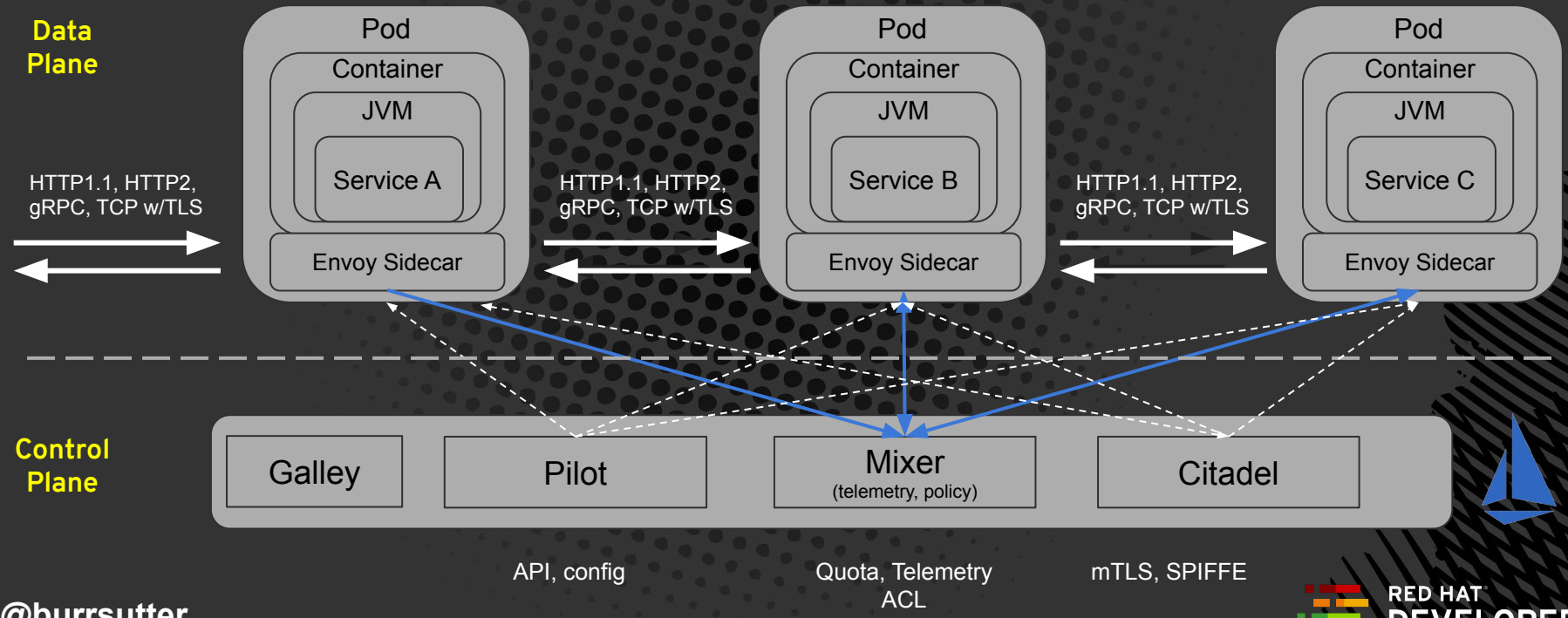
RED HAT DEVELOPER

Envoy is the current sidecar



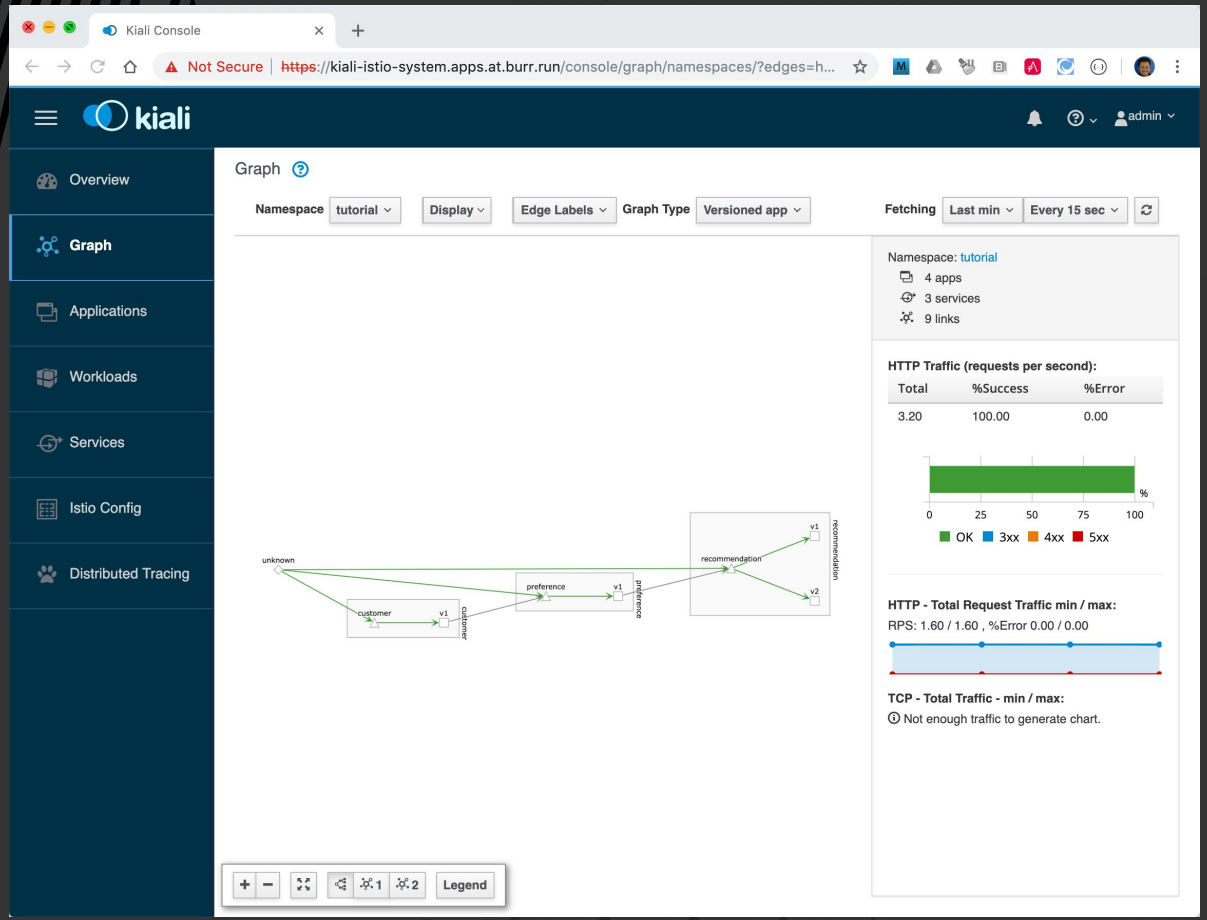
RED HAT DEVELOPER

Istio Data Plane vs Control Plane



@burrsutter

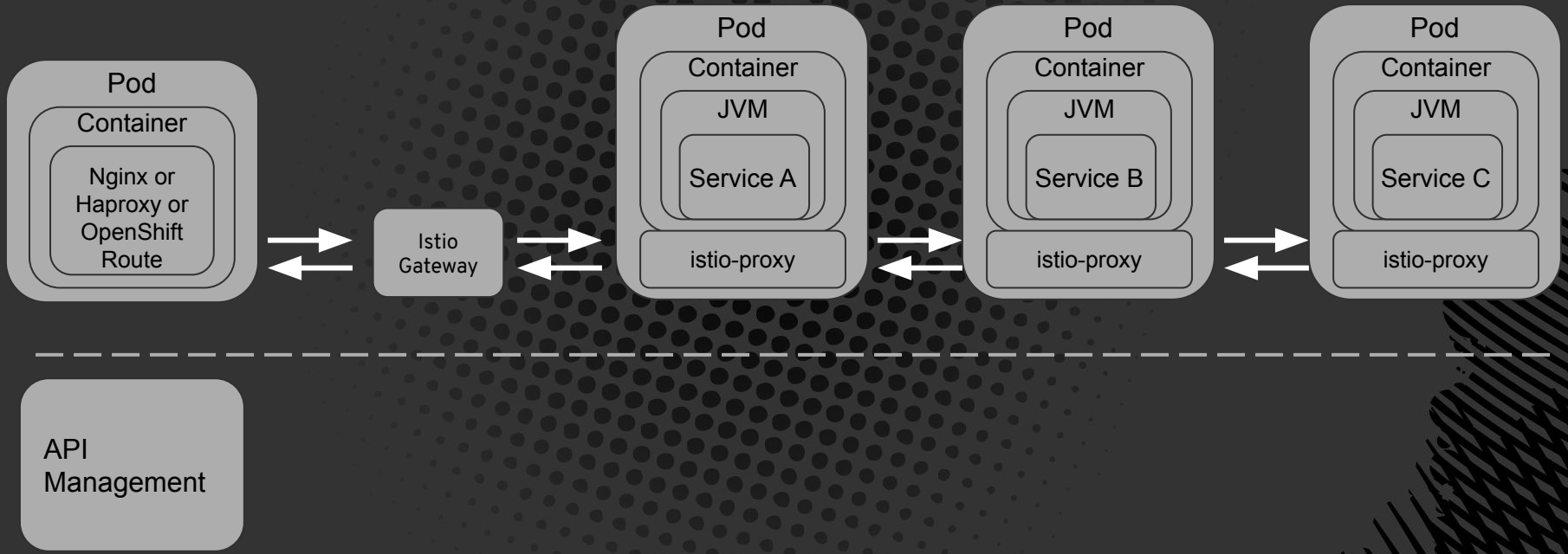
RED HAT DEVELOPER



@burrsutter

RED HAT DEVELOPER

API Gateways



RED HAT
DEVELOPER

Showing off is the fool's idea of glory.

— *Bruce Lee*

RED HAT® DEVELOPER

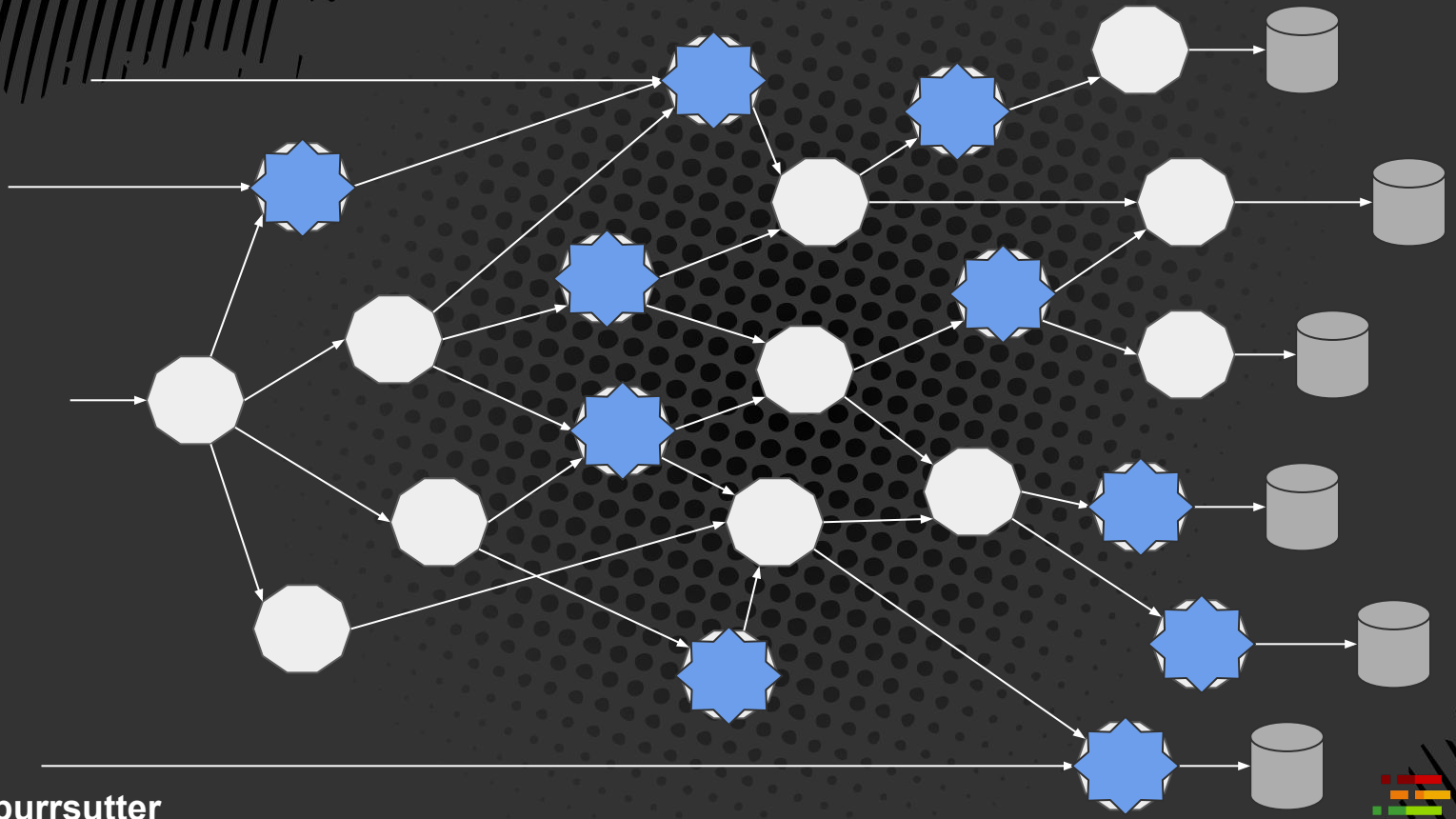
RED HAT
DEVELOPER

"Be happy, but never satisfied.

— Bruce Lee

RED HAT® DEVELOPER

Let there be Functions?



@burrsutter

RED HAT® DEVELOPER

The first question is “is there a suitable service I can consume?” before “is there something I can buy and set up using a cloud provider?”.

Thoughtworks Nov 29 2017

RED HAT® DEVELOPER

It is Serverless because of BaaS/SaaS

(managed by another party services).

RED HAT DEVELOPER

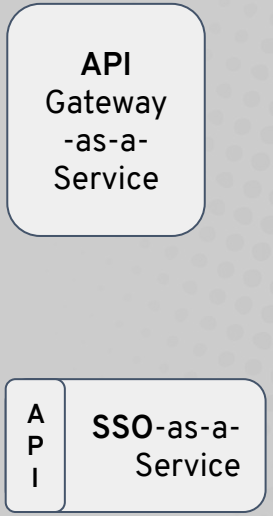
It is all about the Services

RED HAT DEVELOPER

HTTP Input/Output Service

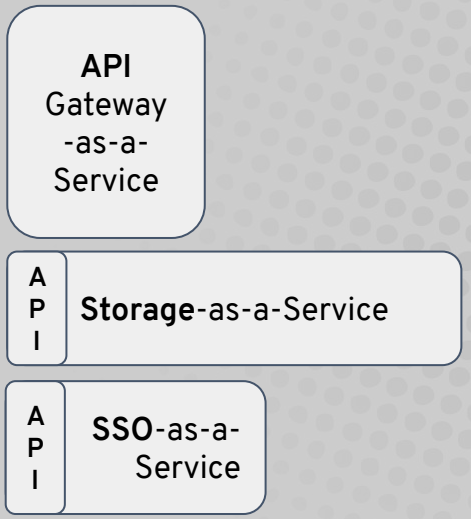
RED HAT
DEVELOPER

Authentication Service



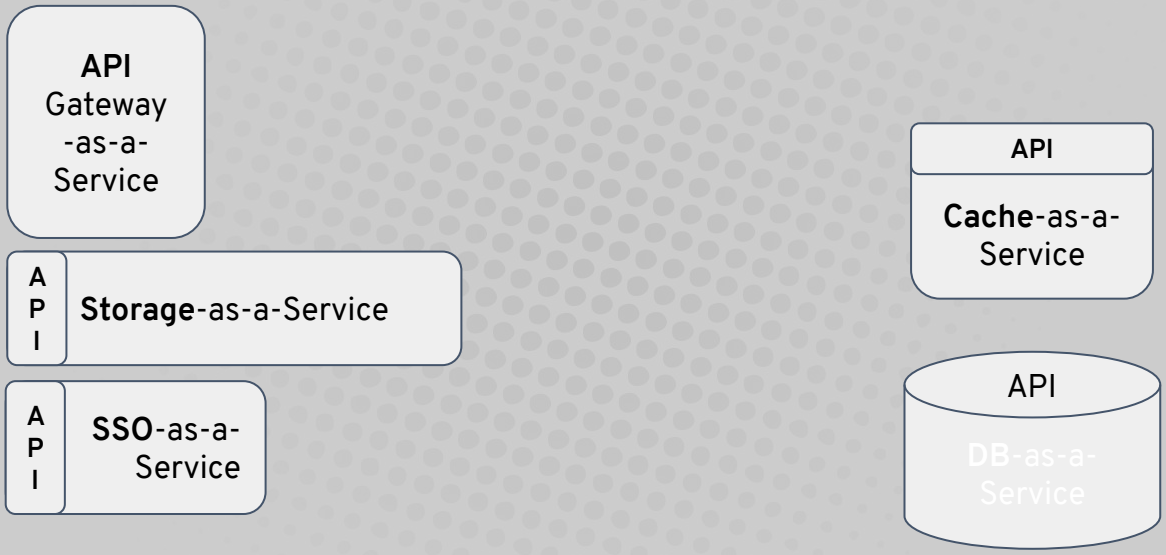
RED HAT
DEVELOPER

File Storage Service



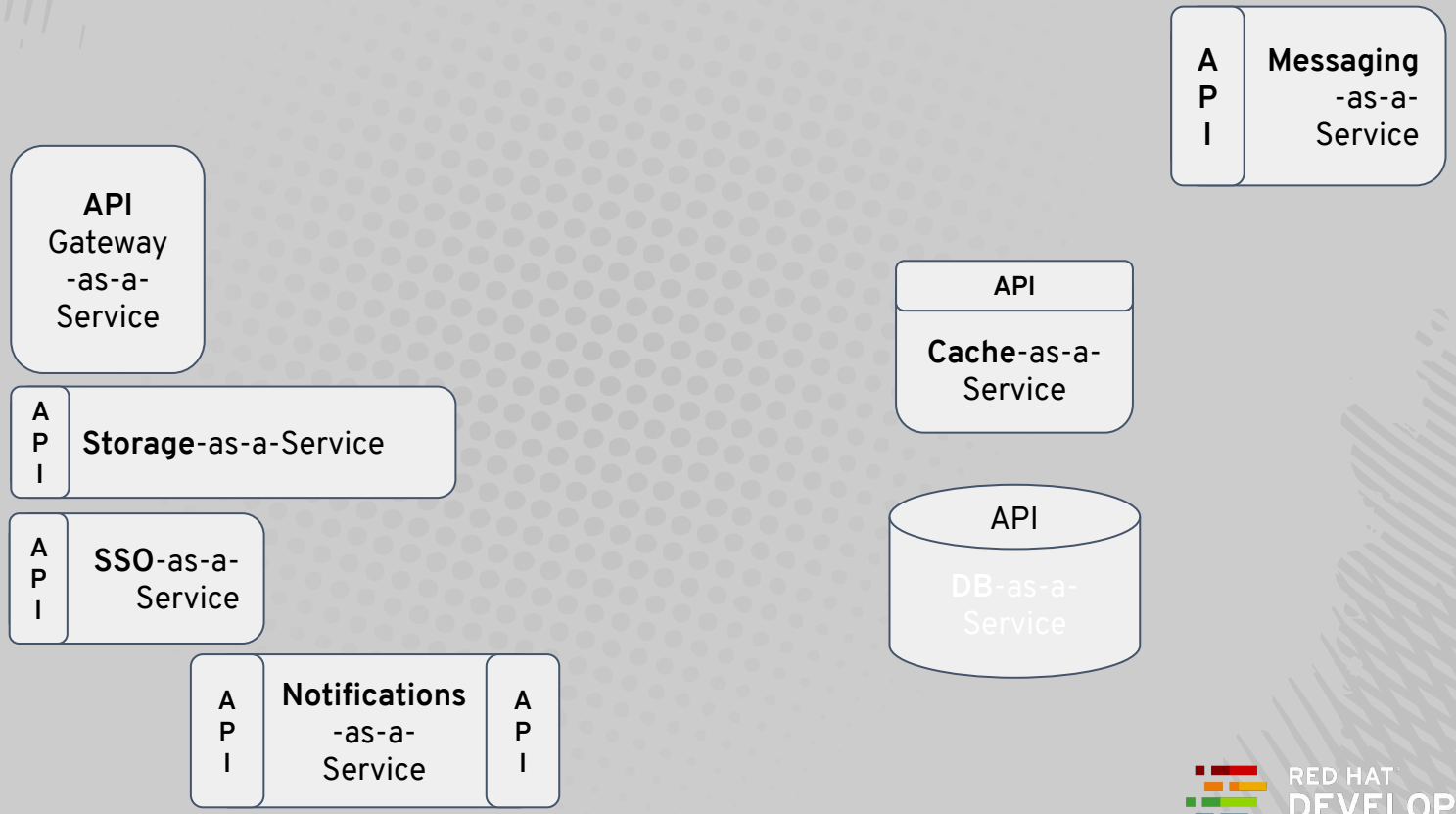
RED HAT DEVELOPER

Data Services



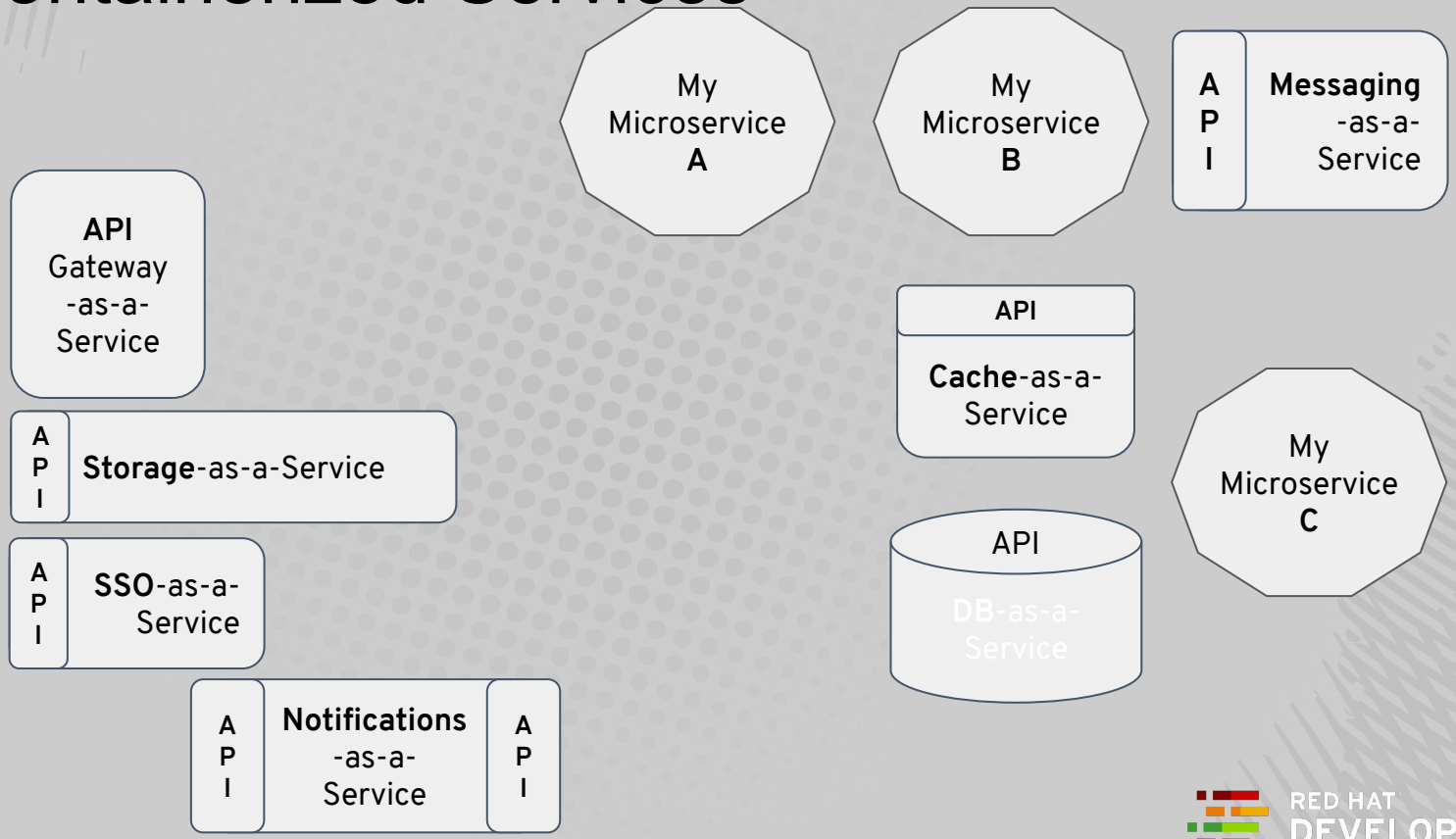
RED HAT
DEVELOPER

Connectivity Services



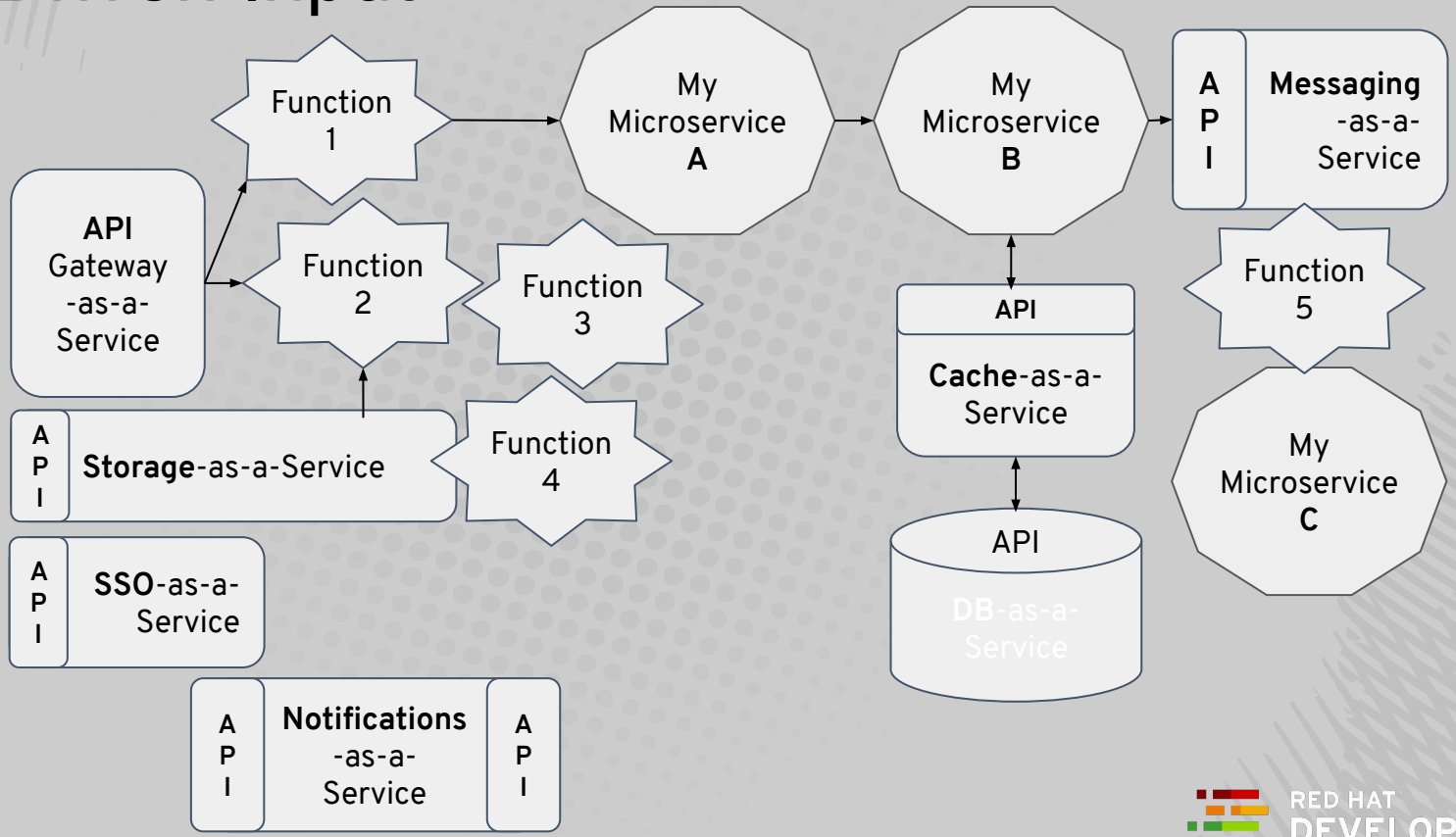
RED HAT DEVELOPER

Your Containerized Services



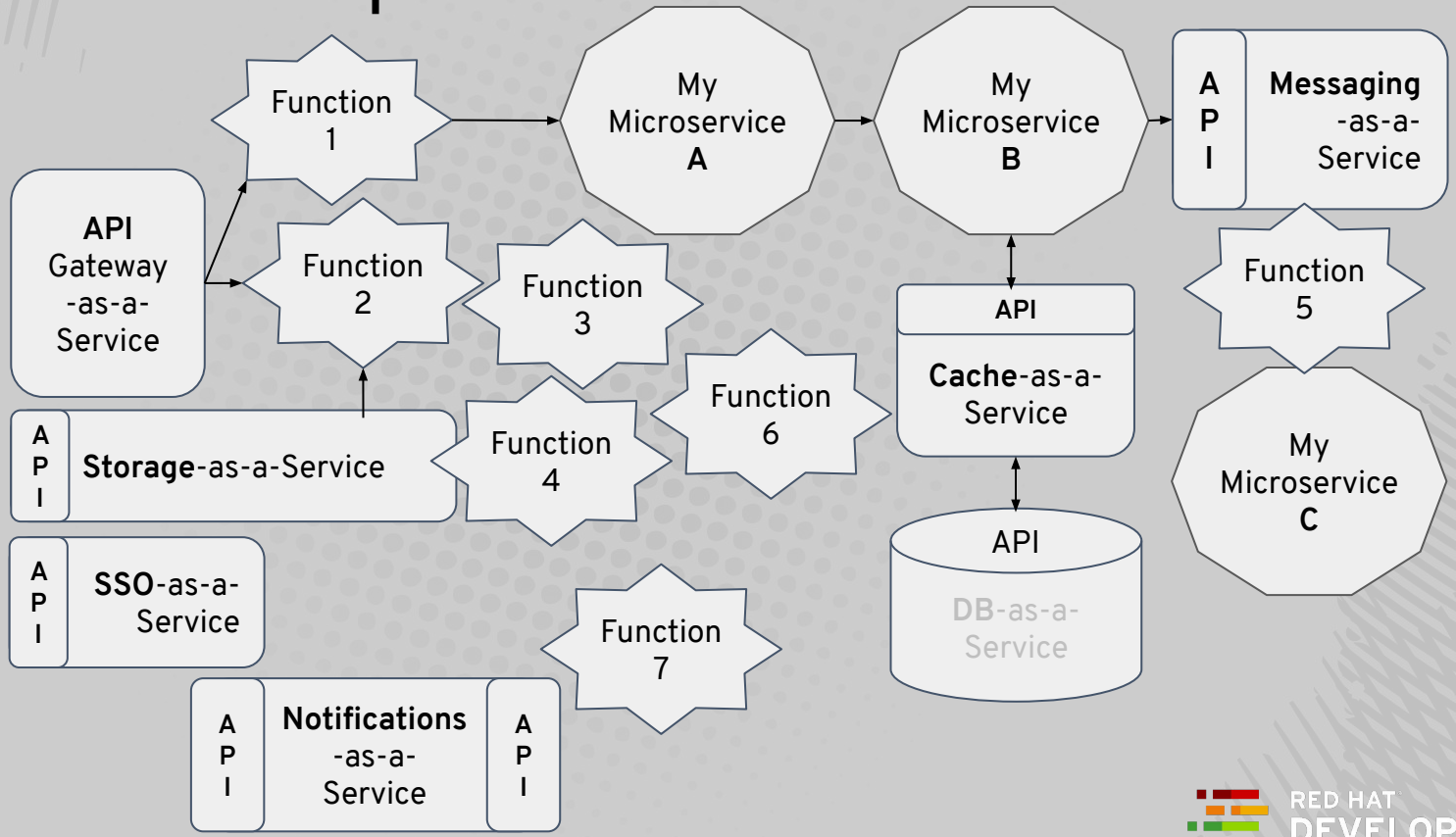
RED HAT DEVELOPER

Event-Driven Input



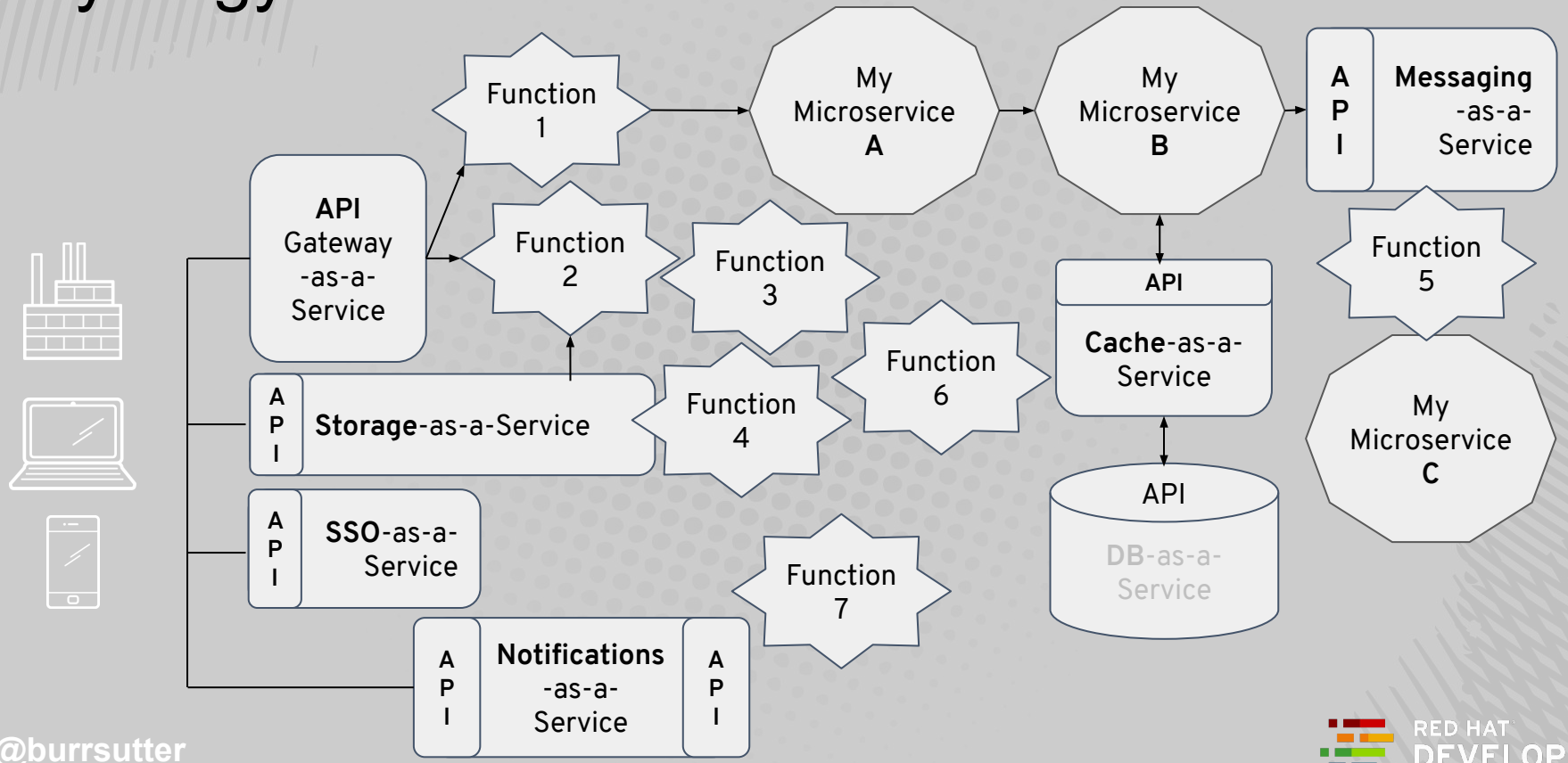
RED HAT DEVELOPER

Event-Driven Output



RED HAT DEVELOPER

Synergy



RED HAT DEVELOPER

Supersonic, Subatomic Java

QUARKUS.IO

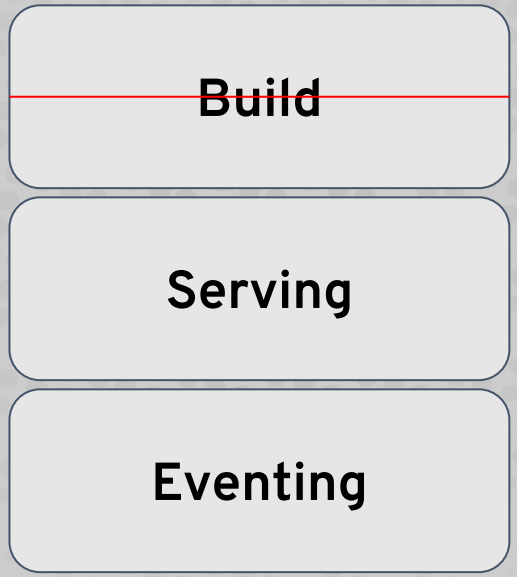
@burrsutter

RED HAT DEVELOPER

Knative

RED HAT DEVELOPER

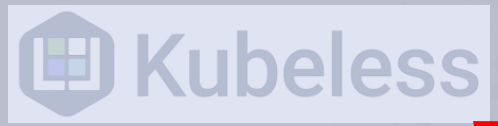
Knative's Primary Components



RED HAT DEVELOPER

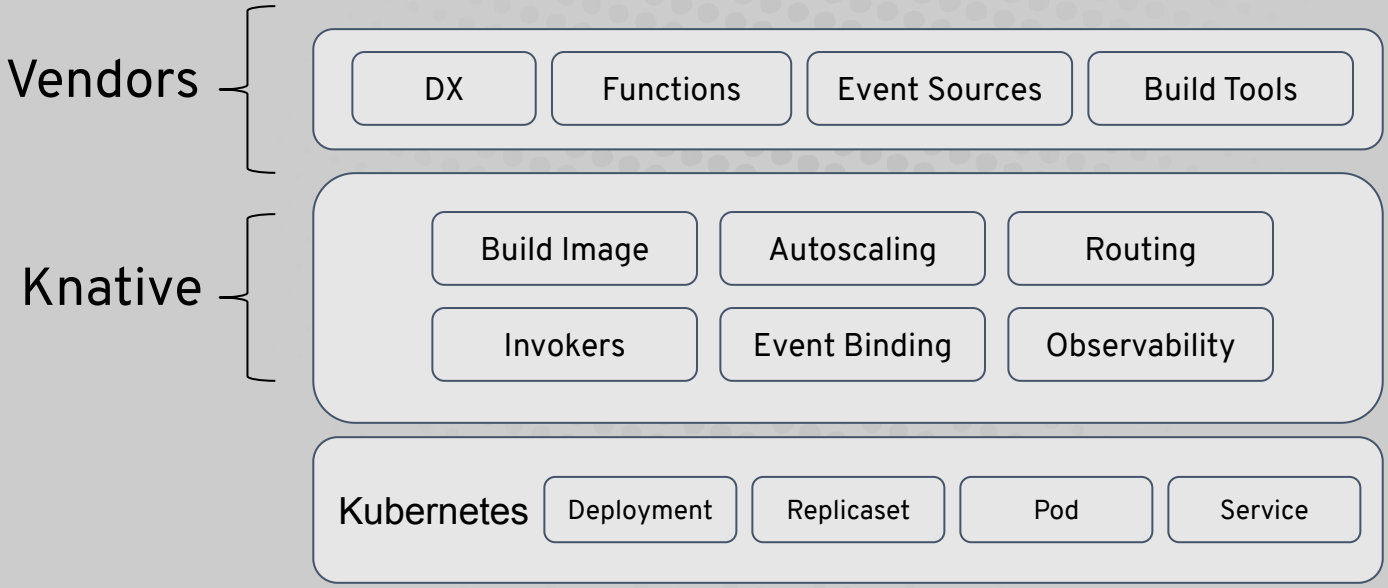
FaaS Kubernetes Players

FaaS Kubernetes Players



Why?

Primitives



RED HAT DEVELOPER

What is Knative?

"Kubernetes-based platform to build, deploy, and manage modern serverless workloads."

"Essential base primitives for all"

"Knative provides a set of middleware components that are essential to build modern, source-centric, and container-based applications that can run anywhere: on premises, in the cloud, or even in a third-party data center"

RED HAT DEVELOPER

Interesting Capabilities

- Scale-to-zero: No pod == no memory
- Scale-from-zero: Traffic spike starts N pods
- Configurations & Revisions - built-in Blue/Green
- In-Cluster Image Building
- Traffic splitting
- Eventing System (Beyond HTTP)

RED HAT DEVELOPER

Knative Serving

RED HAT DEVELOPER

Knative Serving

```
kubectl get crd | grep serving
```

```
configurations.serving.knative.dev
revisions.serving.knative.dev
routes.serving.knative.dev
services.serving.knative.dev
```


RED HAT DEVELOPER

Knative Serving Autoscaler

kubectl -n knative-serving edit configmap config-autoscaler

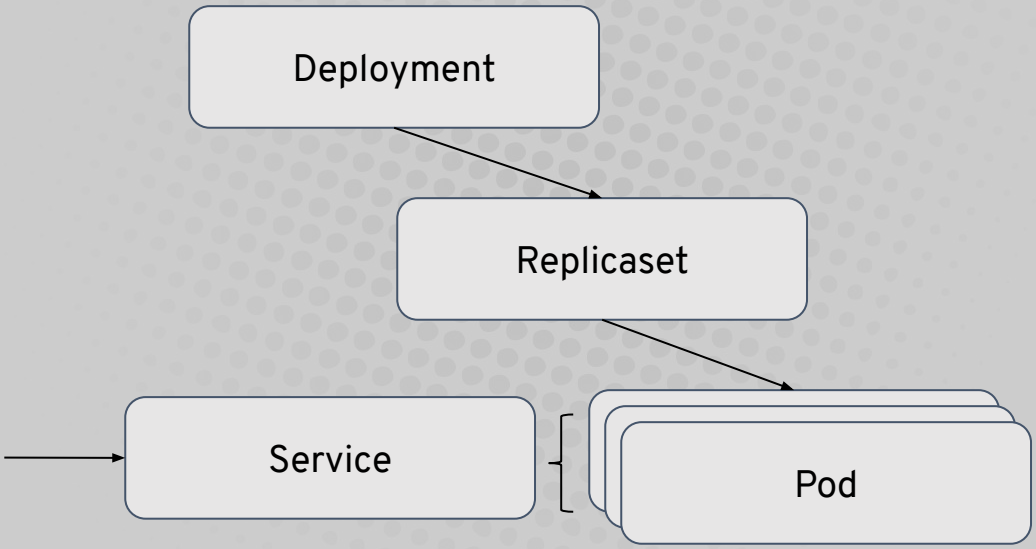
```
container-concurrency-target-default: "1"

scale-to-zero-grace-period: 30s

stable-window: 30s
```

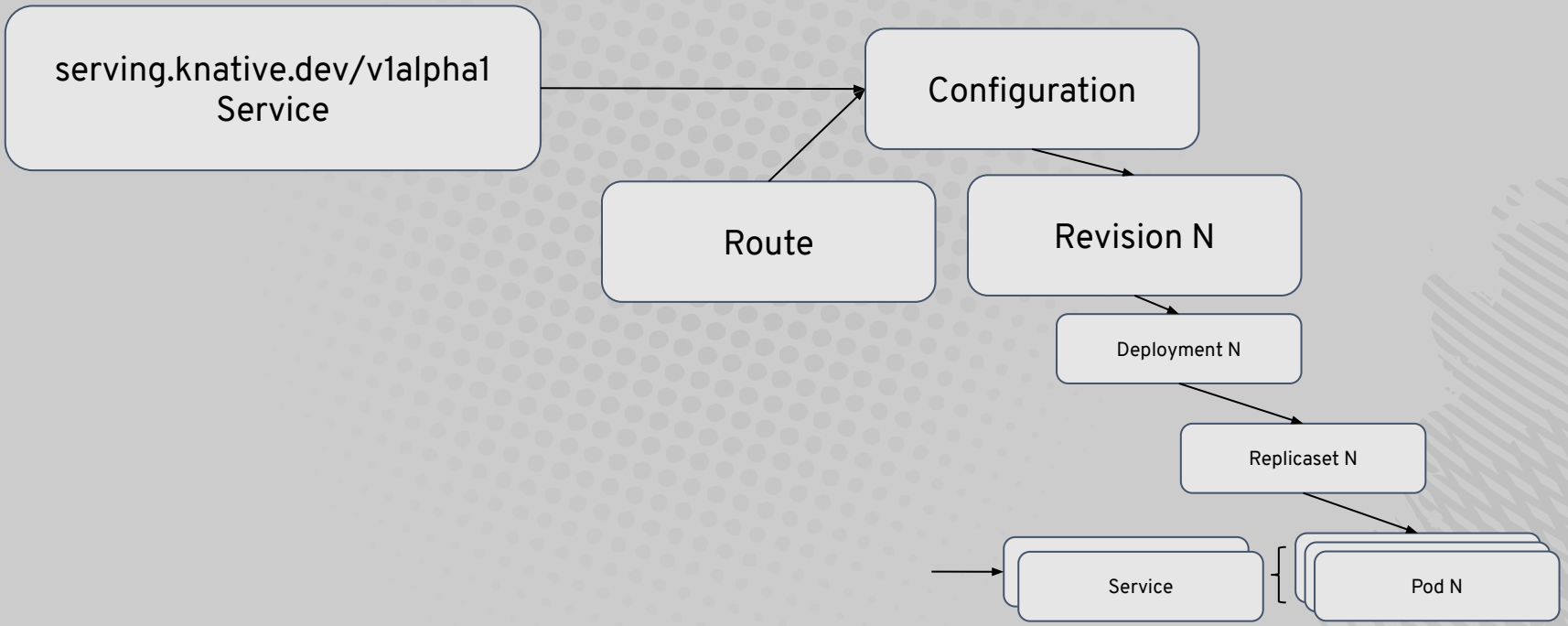

RED HAT DEVELOPER

kubectl apply -f Deployment.yaml

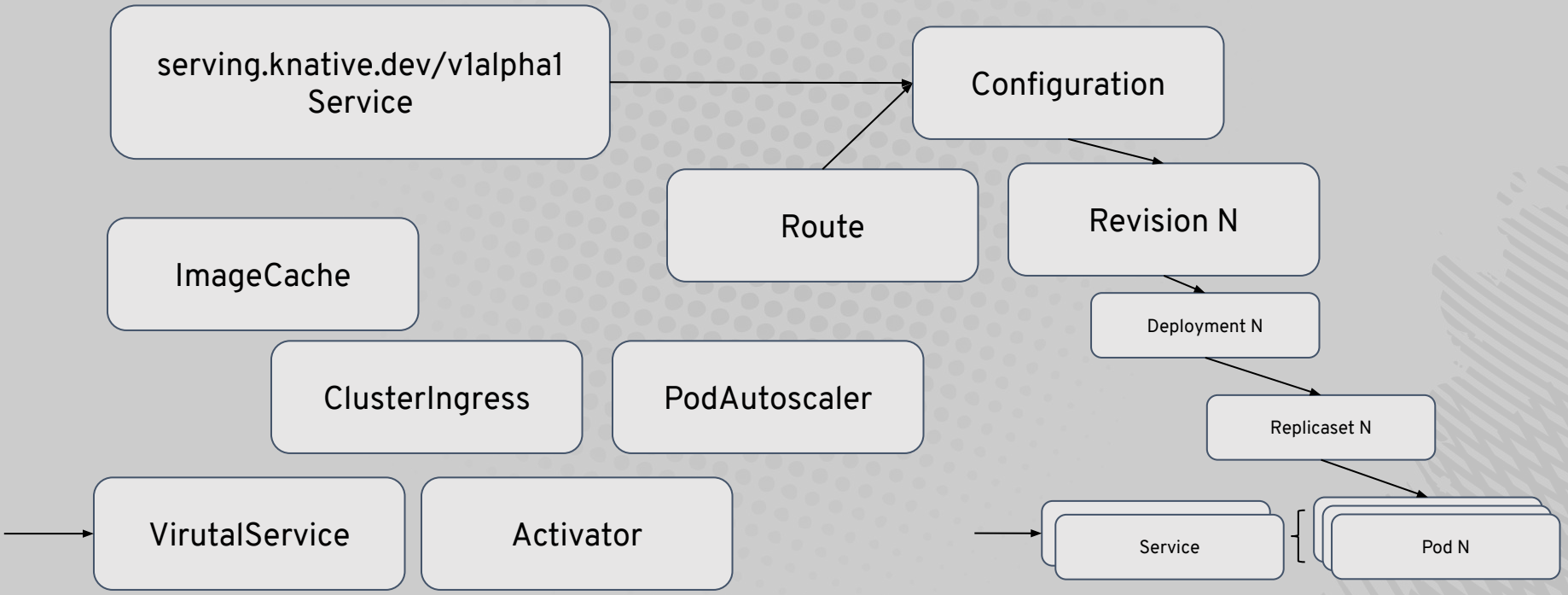


RED HAT DEVELOPER

kubectl apply -f ksvc.yaml



kubectl apply -f ksvc.yaml



Knative Build

RED HAT DEVELOPER

Knative Build

```
kubectl get crd | grep build
```

```
builds.build.knative.dev

buildtemplates.build.knative.dev

clusterbuildtemplates.build.knative.dev
```


RED HAT DEVELOPER

Knative Build Resources

1. Secret	User ID/Password for remote registry
2. ServiceAccount	Associated with the Secret
3. PVC: Maven	Caches Maven downloads
4. PVC: Image Cache	Caches Linux image downloads
5. Build	The Steps
6. Service	Knative Serving Service

RED HAT DEVELOPER

TEKTON

@burrsutter

RED HAT DEVELOPER

Knative Eventing

RED HAT DEVELOPER

Knative Eventing Sources

Github
Websockets
AWS SQS
GCP PubSub
Apache Kafka
Apache Camel

https://github.com/knative/eventing-sources/tree/master/contrib

https://github.com/knative/eventing-sources

RED HAT DEVELOPER

Knative Eventing

```
kubectl get crd | grep eventing
```

```
channels.eventing.knative.dev

subscriptions.eventing.knative.dev

cronjobsources.sources.eventing.knative.dev

githubsources.sources.eventing.knative.dev

kuberneteseventsources.sources.eventing.knative.dev

containersources.sources.eventing.knative.dev
```


RED HAT DEVELOPER

Resources

https://learning.oreilly.com/live-training/ (search for "burr")

bit.ly/9stepsawesome - Introduction to Kubernetes (recording)

bit.ly/istio-tutorial - Istio: Enter the Service Mesh (bit.ly/istio-intro)

bit.ly/knative-tutorial

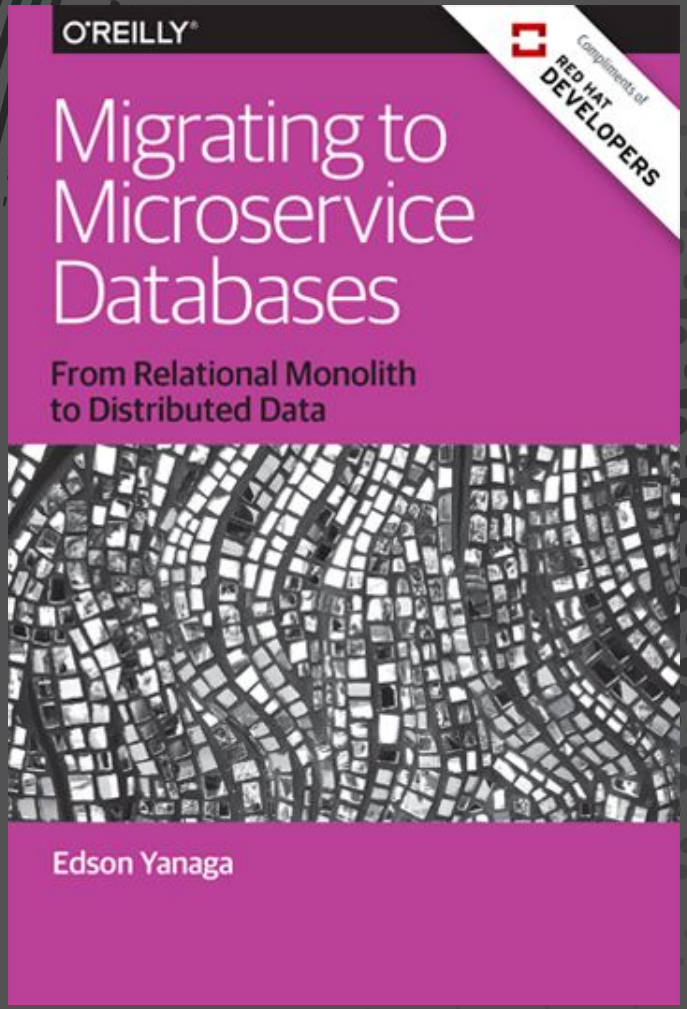
bit.ly/quarkus-tutorial

https://blog.openshift.com/knative-building-your-serverless-service/

https://blog.openshift.com/knative-serving-your-serverless-services/

https://blog.openshift.com/knative-configurations-routes-and-revisions/

RED HAT DEVELOPER

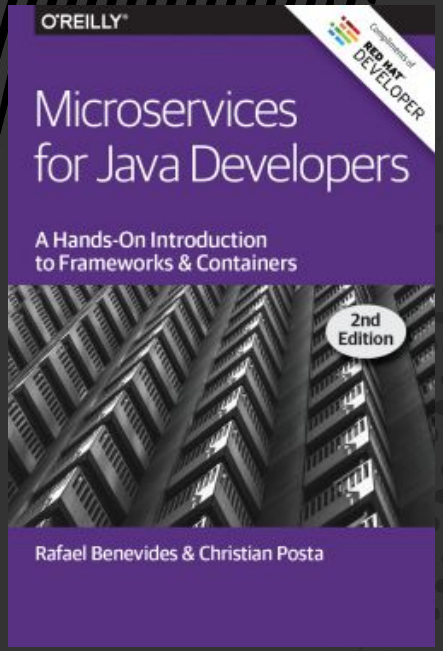


bit.ly/mono2microdb

@burrsutter

RED HAT
DEVELOPER

bit.ly/javamicroservicesbook

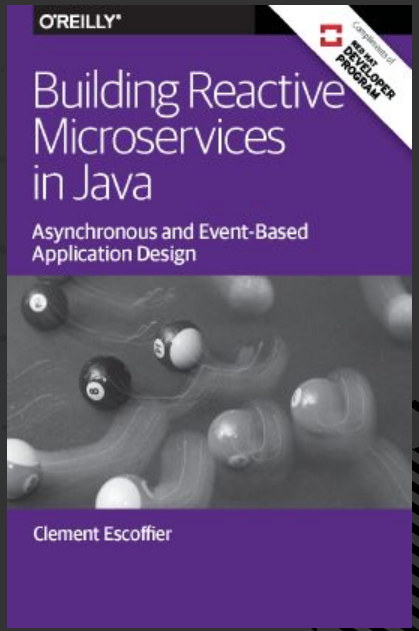


Free eBooks from developers.redhat.com

Microservices Introductory Materials

Demo: bit.ly/msa-instructions
Slides: bit.ly/microservicesdeepdive
Video Training: bit.ly/microservicesvideo
Kubernetes for Java Developers
9 Steps to Awesome with Kubernetes

bit.ly/reactivemicroservicesbook



Advanced Materials

bit.ly/istio-tutorial
bit.ly/knative-tutorial
learn.openshift.com/servicemesh

bit.ly/istio-intro
bit.ly/serverlesskube

@burrsutter

RED HAT DEVELOPER

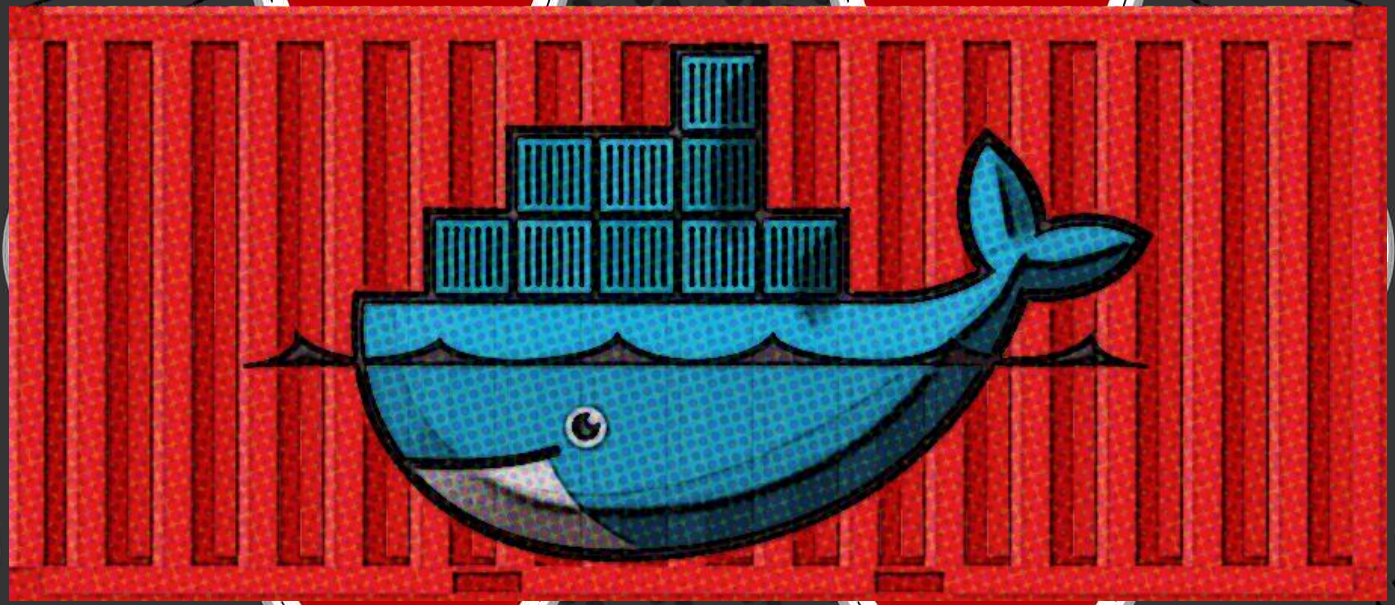
"Don't fear failure.
Not failure, but low aim,
is the crime.
In great
attempts it is
glorious even to fail.

— Bruce Lee

RED HAT DEVELOPER

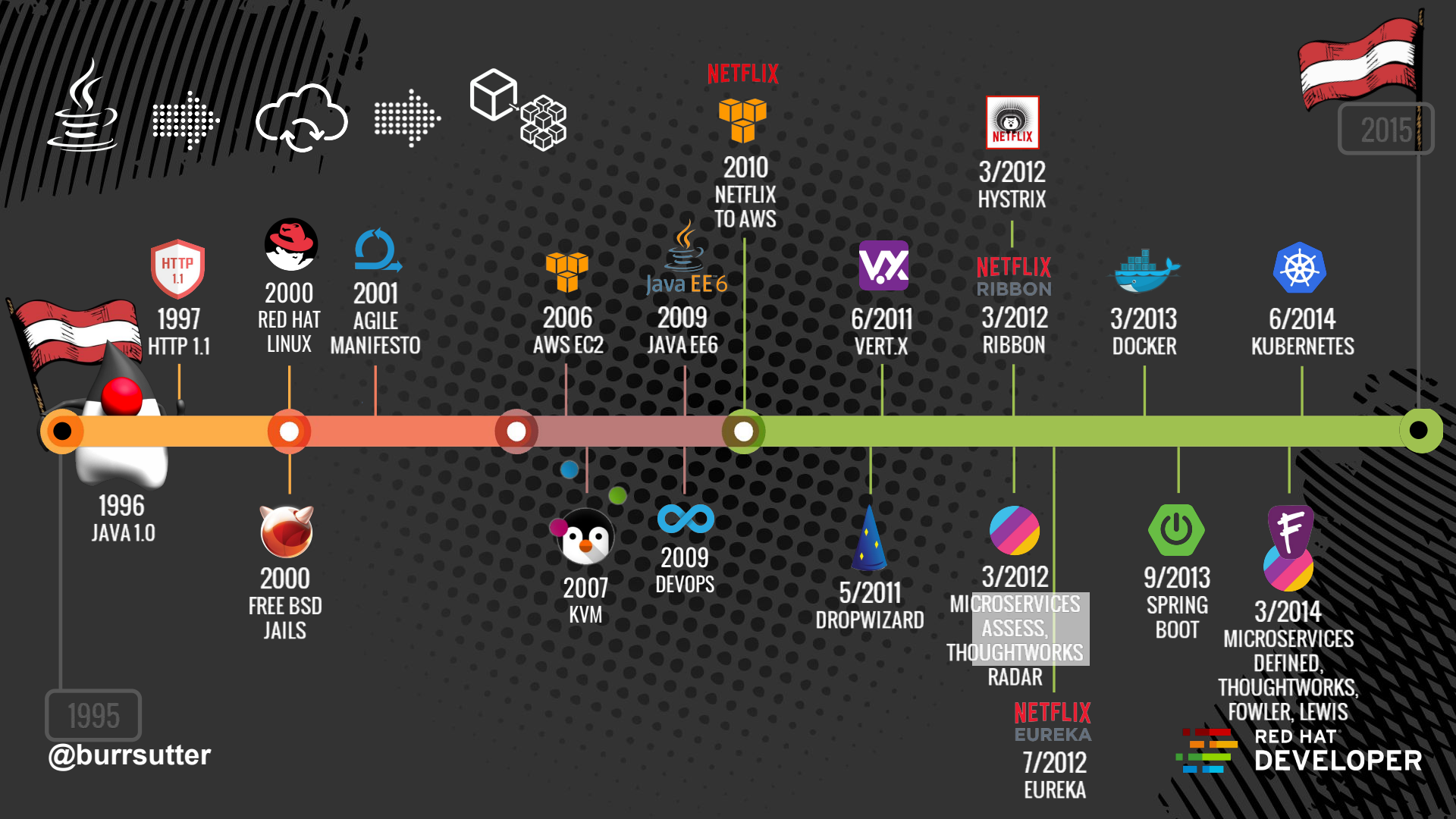
It’s just the beginning ...
The End
bit.ly/RigaBurr2019

MEANWHILE, IN CONTAINER-LAND ...
@burrsutter
RED HAT
DEVELOPER

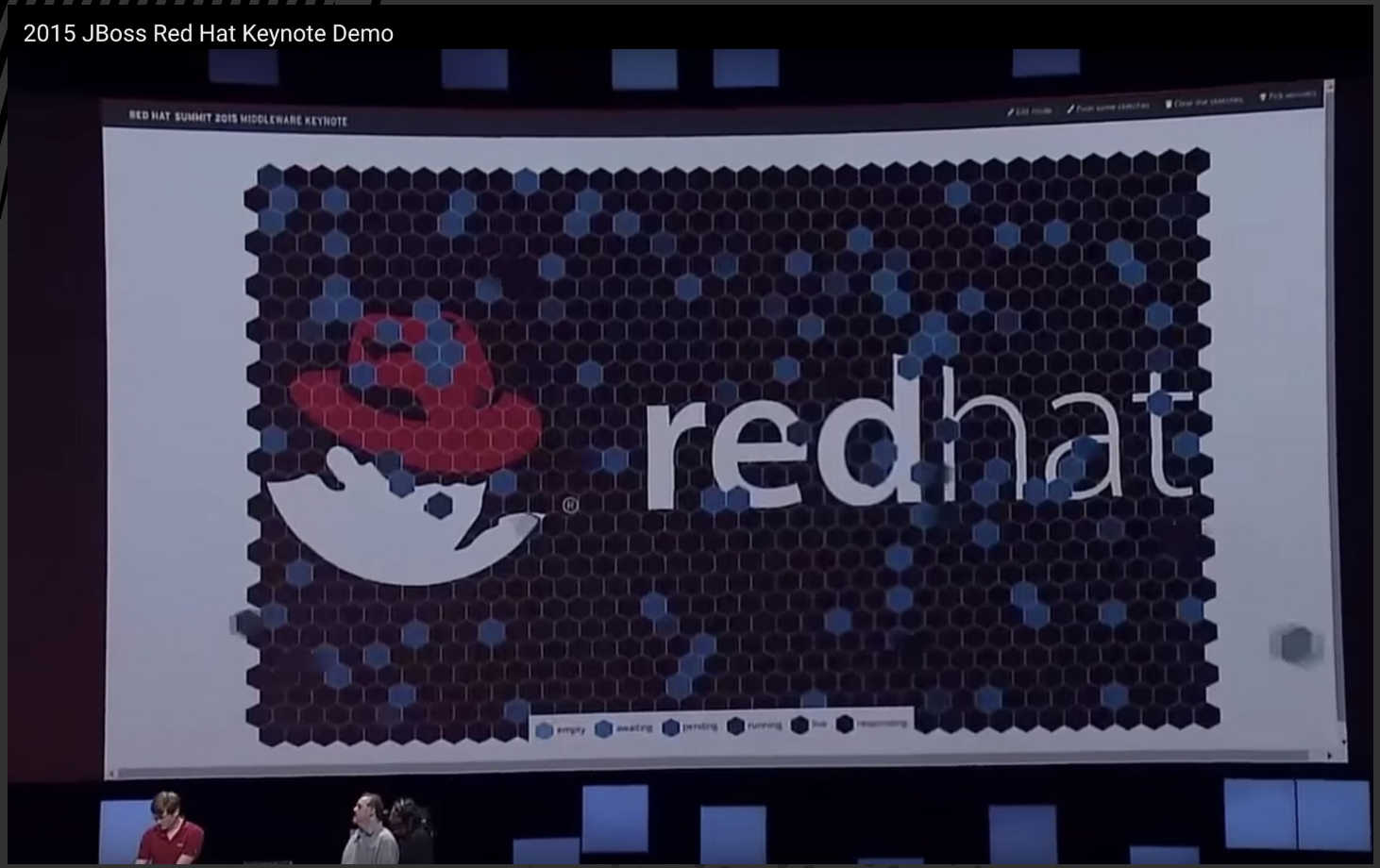


@burrsutter

RED HAT DEVELOPER


1995
1996
JAVA 1.0
1997
HTTP 1.1
2000
RED HAT
LINUX
2000
FREE BSD
JAILS
2001
AGILE
MANIFESTO
2006
AWS EC2
2007
KVM
2009
JAVA EE6
2009
DEVOPS
NETFLIX
2010
NETFLIX
TO AWS
6/2011
VERT.X
5/2011
DROPWIZARD
3/2012
HYSTRIX
NETFLIX
RIBBON
3/2012
RIBBON
3/2012
MICROSERVICES
ASSESS,
THOUGHTWORKS
RADAR
NETFLIX
EUREKA
7/2012
EUREKA
3/2013
DOCKER
9/2013
SPRING
BOOT
6/2014
KUBERNETES
3/2014
MICROSERVICES
DEFINED,
THOUGHTWORKS,
FOWLER, LEWIS
2015
@burrsutter
RED HAT
DEVELOPER

2015 JBoss Red Hat Keynote Demo



2015

Launch 1000+ Containers

Audience Claims a Container

@burrsutter

https://www.youtube.com/watch?v=GCtpncA0Ea0&feature=youtu.be&t=1031

RED HAT® DEVELOPER

Supersonic, Subatomic Java

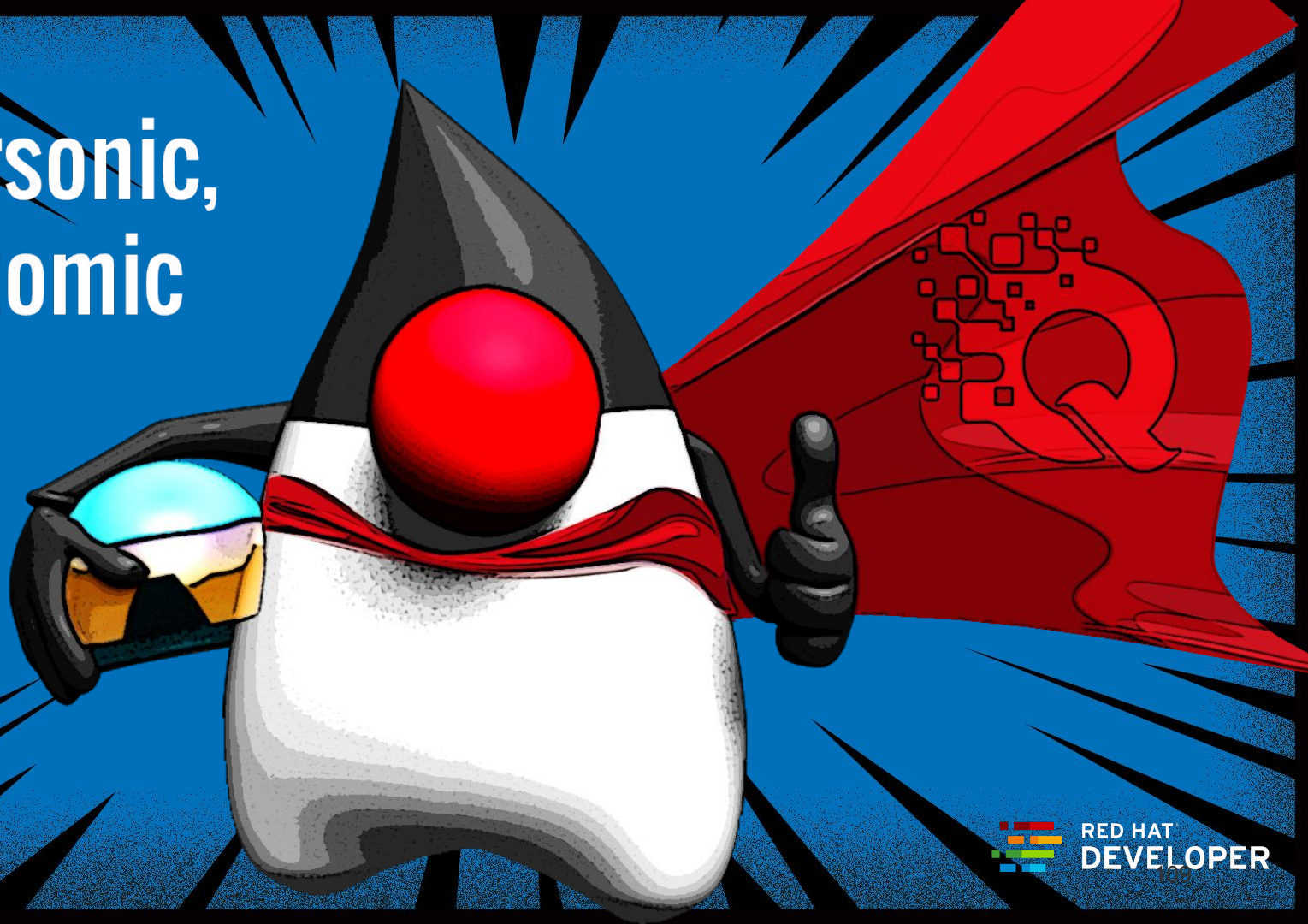
QUARKUS.IO

@burrsutter

RED HAT DEVELOPER

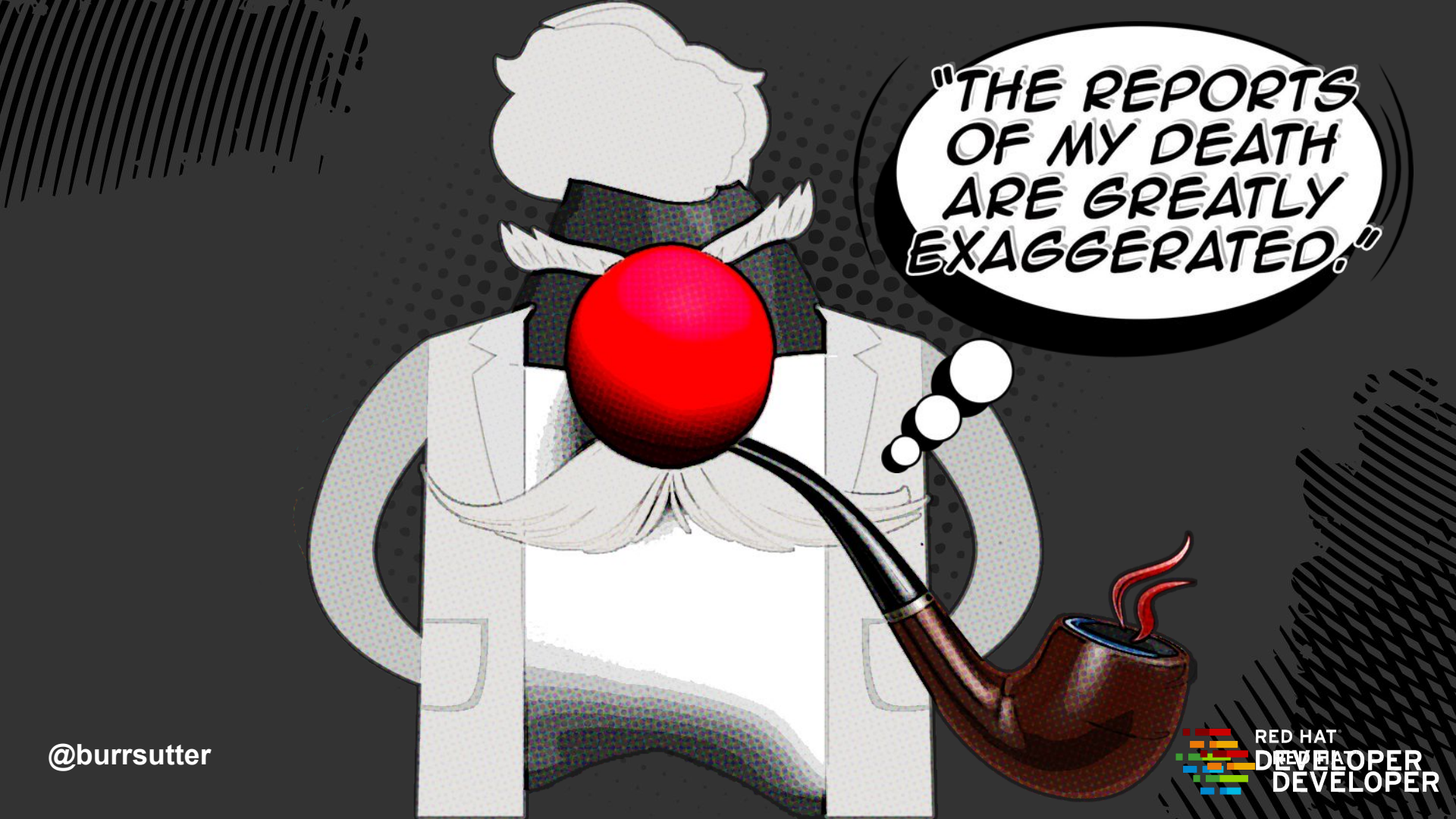
Supersonic, Subatomic Java

QUARKUS.IO



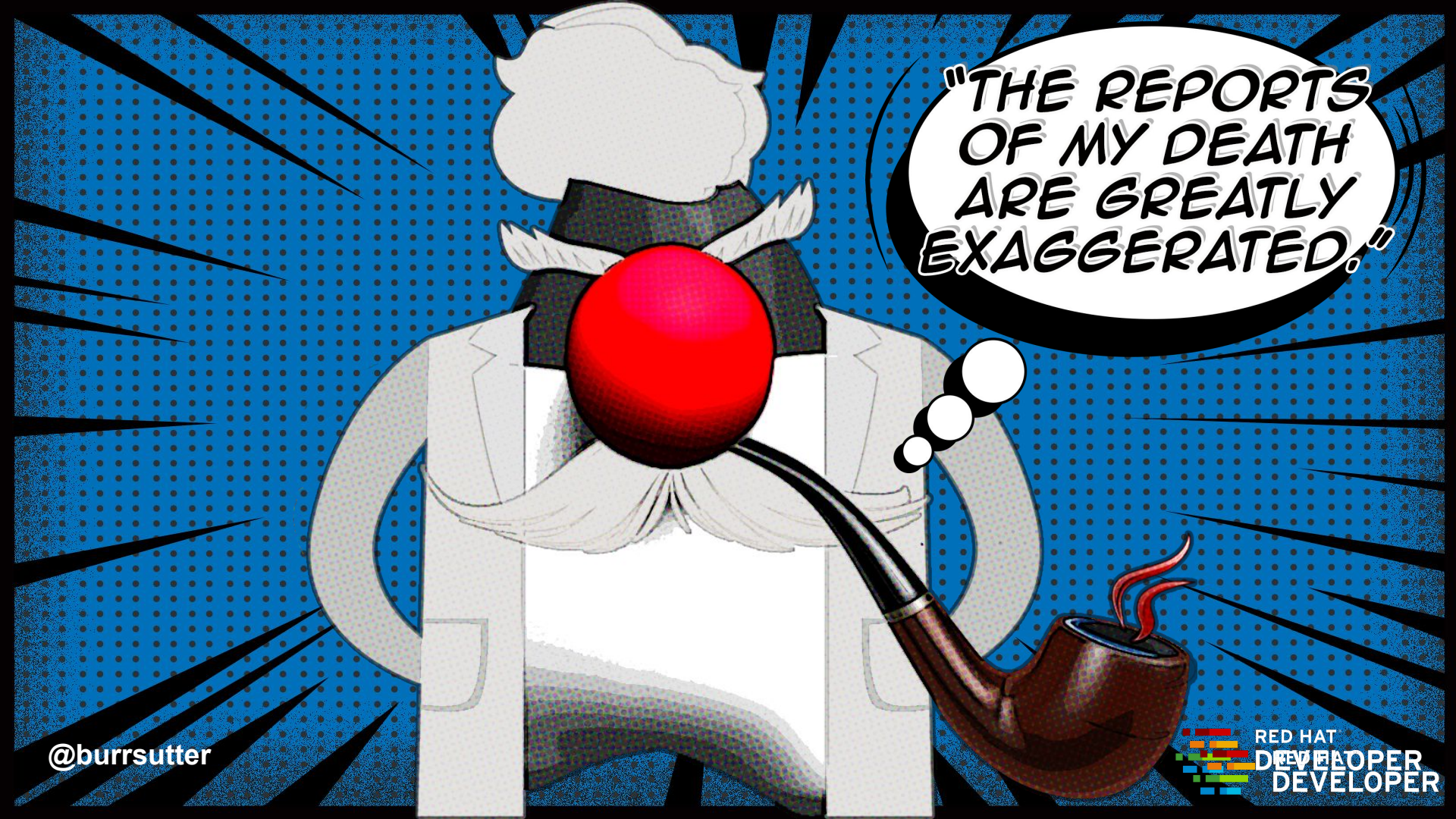
@burrsutter

RED HAT DEVELOPER



@burrsutter

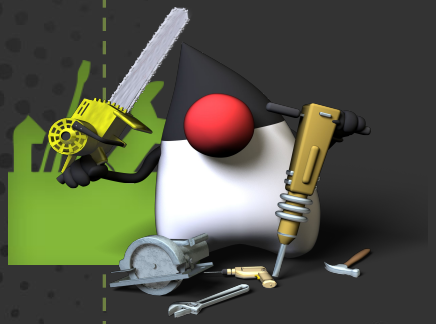
RED HAT DEVELOPER


"THE REPORTS OF MY DEATH ARE GREATLY EXAGGERATED."
@burrsutter
RED HAT DEVELOPER

Still, so much goodness ...

@burrsutter

RED HAT DEVELOPER



@burrsutter

RED HAT® DEVELOPER

Meet Quarkus

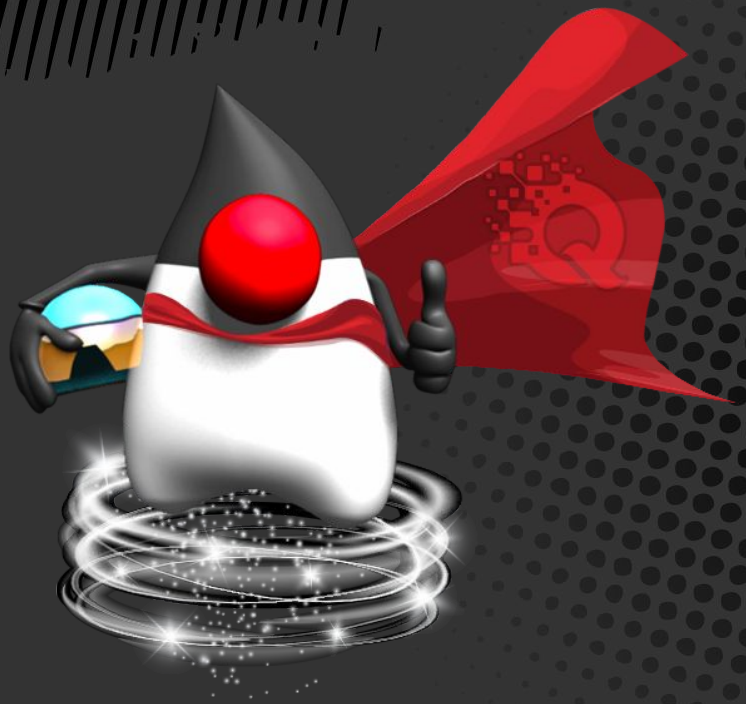
Supersonic Java

developer

Enough ^ love to go around.

@burrsutter

RED HAT DEVELOPER



Live reload

Imperative and reactive

Serverless

Microservices

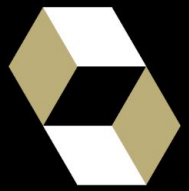
Fat jars and native executables

Optimized for JAX-RS & JPA (Hibernate) ...

@burrsutter

RED HAT DEVELOPER

ECLIPSE VERT.X



HIBERNATE

RESTEASY

APACHE CAMEL

ECLIPSE MICROPROFILE

NETTY

KUBERNETES

OPENSHIFT

JAEGER

PROMETHEUS

APACHE KAFKA

INFINISPAN

@burrsutter

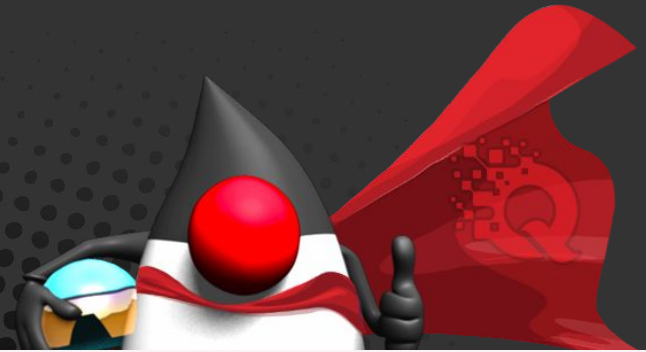
RED HAT® DEVELOPER

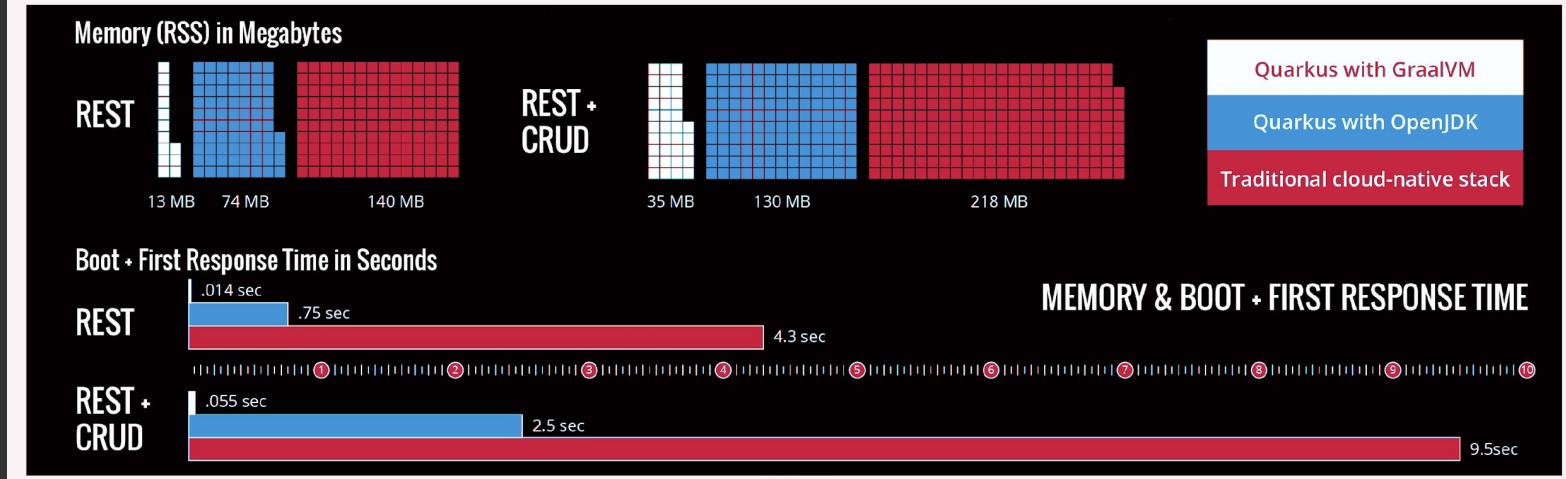
Kubernetes Native

@burrsutter

RED HAT DEVELOPER

RED HAT
DEVELOPER





@burrsutter

RED HAT DEVELOPER